Application of the Wrapper Framework for Robust Image Segmentation For Object Detection and Recognition

Michael E. Farmer
University of Michigan-Flint
USA

1. Introduction

Traditional methods for object detection and classification in images involve either matched filter detectors which are convolved over the image, or else strong image segmentation followed by classification of the resultant segmented regions in the image. Neither of these have lived up to their potential due to (i) the inflexibility of the first approach in detecting objects of varying scale and orientation in varying collection conditions, and (ii) the inherent semantic gap between segmentation and classification in the second approach. Existing segmentation algorithms are built upon the following two common underlying assumptions; (i) the object homogeneity with respect to some characteristic, and (ii) difference between adjacent regions. In this chapter, we propose improving the segmentation process by infusing semantic knowledge into the segmentation process by combining the problems of segmentation and classification through a wrapper framework. Li et al. have noted, "it is often difficult....to determine which regions....should be used for the final segmentation" (Li et al., 2000). The goal of the wrapper framework is to directly address this problem by integrating segmentation processing with a classification process to provide the required semantic information needed to identify the regions of interest.

The key to integrating semantics into the low-level segmentation is to utilize additional feature information of the objects of interest to provide the additional needed contextual information. The features available for classifying an object for image retrieval include texture, color, shape and structure (Safar, 2000). Since texture and color are used as low-level cues, shape and structure are the remaining features to provide additional semantic content. Using the structure of the objects of interest requires associating regions in the image with key structures of the object of interest, and then combining these semantically meaningful regions to provide the complete semantics of the desired object. These regions can either be semantically meaningful on their own, for example, head, limbs, torso for recognizing people, or they can be shape fragments that consistently occur on an object, for example using critical object boundary characteristics. A region combination algorithm then uses a shape template of the object of interest to guide the assembly of these fragments.

We believe a fundamental flaw of these structure-based approaches is requiring the identification of critical shapes and even semantically meaningful sub-shapes within the image. In images with complex natural illumination shadows and bright regions can be created which obfuscate sub-structures. In order to not require mapping of image regions with sub-structures the wrapper framework uses the *overall* shape of the object of interest as the source of semantic information and does not rely on sub-structures. The approach performs a low-level segmentation of the image, and then, irrespective of the shape of the labeled regions in this segmentation, applies an algorithm to combine regions based on knowledge of the shape of the desired object of interest. The proposed approach has been validated through successful demonstrations on a wide range of image applications including automotive occupant sensing, breast cancer detection in mamograms, and wide area disaster surveillance using aerial imagery.

2. Related Work

There is an abundance of literature on image segmentation, due to its importance in serving as the foundation for applications such as image understanding, object detection, and content-based image retrieval. Unfortunately, mechanisms to improve the results to provide strong segmentation where the objects of interest are reliably isolated from the background has continued to elude researchers. Early methods for improving segmentation involved pixel-level post-processing of the initial segmentation to further regularize the segmentation output. This approach has often relied on mathematical models such as Markov Random Fields (Bouman& Shapiro , 1994) (Kim, et al., 2000), or other models such as the harmonic oscillator model by Shi and Malik (Shi& Malik, 2000). More recently Luo and Guo proposed regularization at a region level rather than a pixel level, and they apply a Markov Random Field to combine regions using a non-purposive grouping approach that combines regions based on a defined characteristics of a 'good' segmentation rather than relying on any model of the desired object of interest (Luo & Guo, 2003).

There has also been a significant amount of research in adaptive image segmentation, where the control parameters of the underlying segmentation algorithm are modified, based on some general figures of merit of the output segmentation (Bhanu & Fonder, 2000). More recent low-level segmentation approaches proposed a continuously executing algorithm where the user stops the algorithm when the resultant segmentation appears acceptable (Tu & Zhu, 2002). These methods still relied on the assumption that the pixels belonging to the object of interest share a common set of low-level image attributes, thereby allowing the object to be extracted as a single entity. Unfortunately even relatively simple objects of interest can be composed of multiple regions of differing texture or color which would cause the object to be oversegmented and hence divided into multiple regions. The results of these approaches had limited generalized performance and demonstrated the need to devise a means for integrating additional semantic information into the segmentation process.

One of the earlier approaches to integrating segmentation with classification for infusing semantic information, involved adjusting the segmenter control parameters of the underlying segmentation algorithm based on the classification of the binary (foreground-background) segmentation (Bhanu & Peng, 2000). Unfortunately, this approach still assumed the object of interest is homogeneous in the segmentation feature, and finding it was a matter of discovering the correct control parameter via the classification results.

Integrating semantics into the segmentation have found some early success in very focused domains, such as the work by by Tu, et al. (Tu et al., 2003) which performs simultaneous human face and word segmentation from an image for a system for assisting the blind. This method directly relies on the fact that the objects of interest can be completely defined by their texture (text) or color (human face). Another approach for integrating classification information into the segmentation process was proposed by Sifakis, et al. where they provide context by providing a set of two coarse object contours, one ensured to be outside of the object of interest, and the other designed to be inside the object of interest, in a manner similar to the marker-based approaches to Watershed processing (Sifakis et al., 2002). While this method clearly provides strong context, it still is based on two key assumptions; (i) the object of interest "should be uniform and homogeneous with respect to some characteristic", and (ii) "adjacent regions should be differing significantly" (Bhanu et al., 1995). Additionally these methods operate at the pixel level and hence are computationally intensive.

One key development in image segmentation has been the developing interest in operating at the region level of images rather than at the pixel level. Some of the earliest work in region-based analysis is by Belongie et al, in their 'Blob world' system, where images were grouped into regions based on color and texture and then the user defined regions of interest based on these parameters for that to search databases (Belongie et al., 1997). Unfortunately, this approach still relies on regions being of understandable interest to the user . Li et al. have relaxed the limitation of identifiable sub-regions, by using properties such as color and texture of *all* of the regions in the image to attempt to allow image retrieval systems to bypass the segmentation process (Li et al., 2000). One drawback of this approach is that it compares not only the foreground, but also the background regions in the two images to derive the similarity, which can be particularly limiting if the object of interest is considerably smaller than the field of view of the image, or if the object of interest may be present in a wide variety of backgrounds. Jing, et al. have also recognized that region analysis is essential for effective retrieval, but their approach uses region color rather than shape for the retrieval feature (Jing et al., 2004). More recently Athanasiadis, et al. have proposed a region-based simultanous segmentation and detection scheme which relies on two low-level features defined by MPEG-7, namely homogenous texture and dominant color to perform low-level segmentaiton. Based on semantic models of objects of interest, these low-level regions are then merged together based on fuzzy relations associated with semantic information regarding the objects of interest (Athanasiadis et al., 2007).

The research highlighted to this point were based on low-level image attributes, such as color, grayscale, or texture, and clearly these failed to provide adequate semantic content for strong segmentation. Clearly, additional features are required for successful segmentation, and these can be found by refering to the body of research from content-based image retrieval where the spectrum of features available for retrieval have been defined and highlighted in Fig. 1 (Safar et al., 2000). Based on this taxonomy integrated segmentation-classification methods have been recently directed at developing structural models of the desired object and using either tree or graph theory-based techniques to assemble detected regions in the image that may correspond to sub-structures in the object of interest (Yu, et al., 2002) (Borenstein & Ullman, 2002) (Lee & Cohen, 2004) and most recently by Cours and Shi (Cour & Shi, 2007). For example, Borenstein and Ullman have developed an approach which searches for object 'fragments' within the image, where these fragments correspond to key identifiable regions of the object of interest but not necessarily semantically meaningful structures (Borenstein & Ullman, 2002). The fragments are found using a correlation detector approach. Borenstein and Malik developed a top-down mechanism to augment the traditional bottom-up segmentation algorithms similar to how the proposed wrapper framework operates. Their top-down approach first integrates the low-level regions into semantically meaningful parts using shape templates, and then further integrates these components to form the desired object of interest (Borenstein & Makik, 2006). Good results are possible with these approaches when applied to relatively well-formed images with relatively simple backgrounds, however, there are two underlying assumptions of these methods that can limit their broader applicability, namely: (i) they define a particular form for the classification problem, namely using tree or graph distances, and (ii) they build the segmentation using specific identifiable sub-regions in the images (e.g., head, arms, torso, etc. for human segmentation), and then rely on the known syntactic structure of the object of interest to assemble these components. One key drawback to these approaches is that syntactic methods can be sensitive to errors in the low-level segmentation which was concisely state by Datta: “extracting semantically meaningful coherent regions is… very challenging” (Datta et al. 2008). This was also demonstrated by Lee, et al. where the algorithm had problems recognizing human poses if the subjects wore gloves thereby hiding the skin color (Lee & Cohen, 2004).

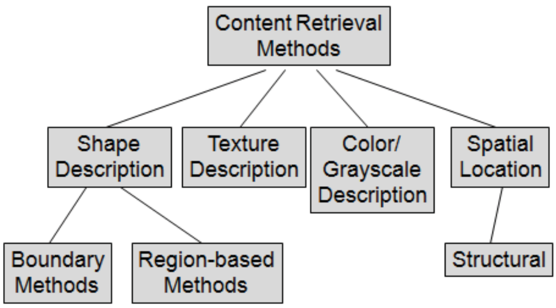
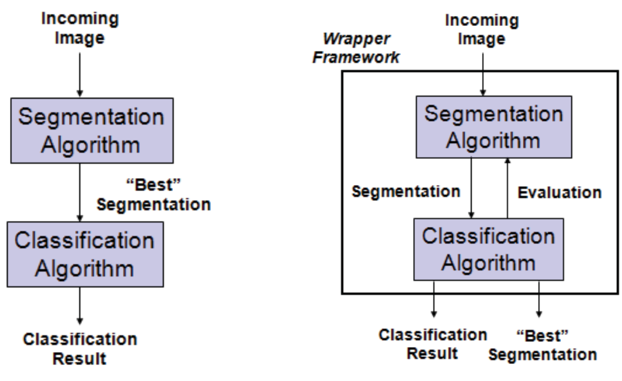


Fig. 1. Taxonomy of feature-based description techniques for image classification.

The approach taken by the wrapepr framework is to use *shape* rather than structure to provide context to the image segmentation problem. Other researchers such as Ko and Byun have added shape rather than structure to their region-based search by adding a small set of shape features to each region (Ko & Byun, 2005). The user then selects from a sample image a number of regions they consider important for query, and the system then computes a combined search distance based on the collection of regions found in the database of images. Like Ko and Byun, the proposed wrapper approach computes shape featues for each region in the image, however rather than relying on region-to-region comparisons, our wrapper approach uses the image classification in a more global scheme. The image classification is used to *assemble* regions derived from the traditional segmentation algorithms rather than simply searching for instances of individual regions. The wrapper approach has numerous advantages over the various methodes described above. One advantage is that by using shape over structure we do not require identifiable sub-structures to be segmented from the background image. Additionally, since we do not rely on particular constructs to represent the problem, such as trees and graphs, it can incorporate any classification algorithm. Also, unlike the fragment approach of Borenstien and Ullman and Borenstien and Malik the wrapper approach is a more organic approach where it builds the desired object from the regions provided in the image rather than *a priori* defined representative sub-images that are searched for in the image. This has an important consequence in that the wrapper approach can utilize any existing image segmentation algorithm to create the regions with which it then operates. Thus rather than being considered another segmentation algorithm, the wrapper approach is actually a *framework* within which any segmenter and classifier can be considered for integration to address the particular problem being addressed.

3. Wrapper Approach to Integrating Segmentation & Classification

We derive the motivation for our approach from the domain of feature selection in pattern recognition, where there are two common mechanisms for selection, namely the filter method and the wrapper method (Dash & Liu, 1997). Filter methods analyze features independently of the classifier and use some 'goodness' metric to decide which features should be kept. Wrapper methods, on the other hand, use a specific classifier, and its resultant probability of error, to select the features. Hence in the wrapper method, the feature selection algorithm is *wrapped* inside the classifier. Based on this, we propose a new paradigm for image segmentation that follows the wrapper methods of feature selection, where we *wrap* the segmentation and the classification together, and use the classifier as the metric for selecting the best segmentation. Fig. 2 compares the traditional image segmentation approaches with our proposed wrapper-based segmentation approach. The classification algorithm provides both the *semantic context* for the segmentation, as well as a *figure of merit* for the resultant segmentation, based on the classification accuracy for the pattern class under consideration.



(a) (b)

Fig. 2. Approaches to image segmentation, (a) conventional methods and (b) proposed wrapper method.

The general processing flow for the entire wrapper framework can be seen in Fig. 3. The processing is divided two distinct phases, (i) conventional (context-free) segmentation, and (ii) wrapper-based (semantic) segmentation. In the conventional segmentation phase no contextual or semantic information is used and the image is segmented based on traditional low-level homogeneity metrics, typically greyscale or color depending on the application. The second phase is the wrapper segmentation phase where the critical semantic information is integrated via the classifier.

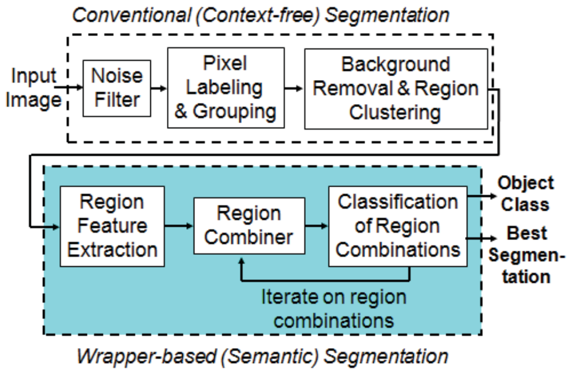


Fig. 3. Processing flow for wrapper-based image segmentation.

3.1 Conventional Segmentation Processing

The conventional segmentation processing flow begins with a low-level order statistic filter such as a median filter for removing high frequency image speckle. Order statistic filters are most attractive since they are edge preserving which prevents degradation of subsequent region labeling due to edge blurring. This step is optional and depends on the quality of the images being processed by the system. The Pixel Labeling and Grouping task in Figure 3 performs conventional low-level weak segmentation. It converts the pixel values into labels, and then groups these labeled pixels into contiguous regions. The series of sub-tasks that comprise the processing of this stage are provided in Fig. 4. The first sub-task is Compute Pixel Data Parameters which is responsible for determining the parameters to be used to determine the low-level pixel labeling based on some common characteristic of the pixel values such as color, grayscale, or texture. There are many mechanisms proposed for defining the 'common characteristics', such as Expectation Maximization (EM), normalized cuts, relaxation methods, region growing methods, and split-and-merge methods, and finally DDMCMC which provides a framework for unifying many of these approaches (Belongie et al., 1997) (Tu et al., 2003). The output of all of these methods is a labeling of the incoming image into a small number of regions. We selected the EM algorithm for the region labeling algorithm was based on its relative ease of use, its flexibility, and its suitability for real-time operation. It fits a mixture of Gaussians that best matches the histogram of the grayscale values. The EM algorithm is attractive because it can easily be extended to use multiple features, such as texture depending on the application.

The Label Pixels task then uses the mixtures defined by EM to label each pixel with its appropriate mixture membership, with typical results being shown in Fig. 5 (b). This labeled image is then mode filtered to further remove isolated pixels. Other regularization algorithms such as the Markov Random Fields methods discussed in Section 2 may be used rather than the mode filter, however, the mode filter is easy to implement, imposes a relatively low processing burden, and has previously been shown to be effective (Farmer & Jain, 2005) (Rabiei et al., 2007).

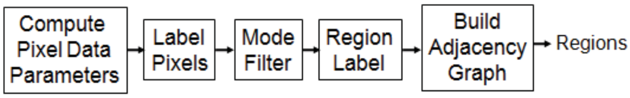
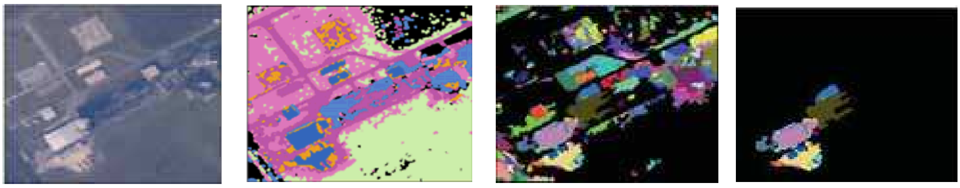


Fig. 4. Processing flow for Pixel Labeling and Grouping**.**

The third sub-task of the Pixel Labeling and Region Grouping Task is Region Label in which the pixels are grouped together into regions of common labeling based on an 8-way connected components algorithm. The Region Label sub-task then removes regions that fall below a user-defined threshold in order to minimize the total number of regions, with the particular threshold value being dependent on the application and the particular input image size. At this point the image has been completely divided into regions of low-level homogeneity (grayscale or color for the applications shown in this chapter). The final sub-task of the Pixel Labeling and Grouping is Build Adjacency Graph. In this sub-task an adjacency graph is constructed to define the relative adjacency of all of the regions in the image. This adjacency graph will play a critical role in subsequent processing for cluster detection and to limit the combinatorial complexity of the region combining algorithm within the wrapper portion of the segmentation process.

Recall from Fig. 3, the next stage in the conventional segmentation processing is the Background Removal and Blob Clustering task. Obviously, the goal of this stage is not to remove then entire background but rather to remove as much background as possible based on simple structural knowledge of images. There will still most likely be significant amounts of background connected to the object of interest, and this remaining background will be removed during region combining. The background in an image is defined as the larger regions and regions along the periphery of the image that typically are not of interest. The size of the background regions is independently defined by two characteristics, the area and the length. Thus regions of large area or large regional extent (such as roads and rivers in surveillance applications) can be ignored. Removing of the background, as shown in Fig. 5 (c), allows the algorithm to now focus on more interesting regions, in a similar manner to human perception where known background regions are ignored while more interesting or ambiguous regions are analyzed further. Once the background is removed, of clusters of regions can readily be detected using the adjacency graph. Clusters are defined by collections of regions that are adjacent to each other, as can be seen in Fig. 5 (d). The wrapper segmentation processing then analyzes each of these region clusters to determine if any objects if interest may be present. The ability of the wrapper framework to process clusters of regions, rather than all the regions in an image, is critical for performance, since the number of possible region region combinations rapidly suffers combinatorial explosion .



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

Fig. 5. Conventional segmentation processing results: (a) example incoming surveillance image, (b) labeled image after mode filter and small region removal, (c) labeled image after background removal, and (d) cluster from lower left region in (c).

3.2 Wrapper Segmentation Processing

Up to this point the input image has been over-segmented to try to maximize the likelihood that the object of interest is not connected to a background region. In order maximize this likiehood, the image is intentionally over-segmented which means the object of interest is most likely sub-divided into multiple regions. The wrapper framework processes each cluster of regions independently and only tests combinations of the regions within each of these clusters, thereby significantly reducing the combinatorial explosion. Clusters consisting of individual region are tested first against a training database to see if any of them may match an object of interest since they require no combining. Then the remaining more complex clusters are processed, where a variety of combinations of region are attempted to see if any of these combinations may match any objects in the training database.

3.2.1 Region Feature Extraction

Recall from Fig. 3, the first task in the wrapper processing is the feature extraction for each region. Fig. 1 demonstrated there are four possible feature spaces for image retrival and classification. The wrapper framework incorporates shape as its semantically rich feature. Shape may be defined by either region or boundary descriptions (Veltkamp & Hagedorn, 2001). While either method can be used to capture the shape of the regions that have been defined as comprising our image, the research to date with the wrapper framework has employed moments to describe these shapes. The geometric moments of an image are defined by (Teague, 1980):

$$M_{lk} = \sum_{j=0}^{N} \sum_{i=0}^{M} I(i,j) i^l j^k , \tag{1}$$

where *I(i,j)* is the value of the image at pixel *(i,j)* and *N* and *M* are the numbers of rows and columns in the image, respectively.

Computing the moments features on every region combination would be computationally prohibitive since many combinations will be generated for every region as will be shown in Section 3.2.2. Fortunately, due to the non-overlapping nature of the regions that comprise the image labeling, the basic geometric moment features can be calculated for each region prior to the subsequent region combining and classification stages of processing. Then during the region combining processing, the moments of the combined regions is simply the sum of the moments of the individual regions, which implies all pixel-level processing need only be performed once, and all subsequent processing is performed at the region level.

This speedup mechanism is related to the concept of Borel sets and the calculation of measures on these sets. A value μ is a measure if it assigns a non-negative number to each subset, which can be seen to be true from Equation (1) since *I(i,j), i*, and *j* are never negative. One important property of these measures is: "if a set is decomposed into a countable number of disjoint Borel sets then the total measure of the pieces equals the measure of the whole", which can be mathematically stated as (Falconer, 2004):

$$\mu\left(\bigcup_{i=0}^{\infty} A_i\right) = \sum_{i=0}^{\infty} \mu\left(A_i\right), \tag{2}$$

where A_i is the i^th subset and μ is the measure. This fact was originally exploited by Spiliotis and Mertzios, where the computation is decomposed into a summation over a set of non-overlapping rectangular homogenous blocks (Spiliotis & Mertzios, 1998). We abstract this approach and, rather than decomposing the image into non-overlapping rectangles, we propose a more natural decomposition of the image into the collection of regions defined by the image labeling. Using the notion of disjoint Borel sets allows representing the image as:

$$I(i,j) = \bigcup_{k=0}^{K} I^{(k)}(i,j)' \tag{3}$$

where $I^{(k)}(i,j)$ is the portion of the image corresponding to region *k*. From this we can now rewrite the geometric moment equation as:

$$M_{lk} = \sum_{j=0}^{N}\sum_{i=0}^{M}\left(\bigcup_{k=0}^{K} I^{(k)}(i,j)\right)\cdot i^l j^k . \tag{4}$$

From Equation (2) we can now replace the measure over the union of subsets as a summation over the individual measures of each of the subsets and obtain:

$$M_{lm} = \sum_{k=0}^{K}\left[\sum_{j=1}^{Nrows}\sum_{i=1}^{Ncols} I^{(k)}(i,j)\cdot i^l \cdot j^m\right]. \tag{5}$$

The term in the square brackets, $\sum_{j=1}^{Nrows}\sum_{i=1}^{Ncols} I^{(k)}(i,j)\cdot i^l \cdot j^m$, is the moment calculation for the moment of order *(l+m)* for the *k^th* region of the image, $I^{(k)}(i,j)$, and we define this moment to be M_{lm}^{k}. We can then rewrite Equation (5) according to:

$$M_{lk} = \sum_{k=0}^{K} M_{mn}^{k}, \tag{6}$$

where we have reversed the order of the summations, and M_{mn}^{k} is the moment of order *(m+n)* corresponding to the *k*^th region. Thus, the geometric moments for the entire image are merely a sum of the geometric moments computed for each region.

Now we can pre-compute the moments for each region, which allows us to add the feature vectors from each region together to compute the moments for any region combination. The

ability to pre-compute features can provide a considerable benefit, since, as Yoshitaka and Ichikawa state: "[feature extraction] processing is one of the most time consuming parts in content-based retrieval. Improving the [feature extraction] processing therefore improves the overall performance... (Yoshitake & Ichikawa, 1998)." We will see in Section 3.2.3 that from this point in the processing all operations will be performed on regions rather than pixels which greatly reduces the overall processing complexity of the wrapper framework.
There are many forms of image moments that can be used for image classifcation, including central, scale invariant, rotationally invariant, legendre, zernicke, et cetera (Teague, 1980). For most image object detection, classification, and retrieval applications central moments are always required since they provide translational invariance within the image.

3.2.2 Region Combining

For Region Combining processing an algorithm is required which will combine subsets of the regions in the image together while assuming a particular object class is present in the image. The wrapper framework operates by assuming a pattern class *C* to be the true class, and computes the classification distance of candidate segmentations to that class. The specifics of the classification algorithm upon which the wrapper relies are provided in Section 3.3. The region combining task is performed for every class, *C*, and at the completion of the processing of all the candidate classes, the class *C* that provides the highest membership probability, $P(\{X_k\}|C)$, is selected, where the set $\{X_k\}$ defines the subset of regions that comprise the best segmentation for iteration *k* of the algorithm. Likewise, the set of regions $\{X_k\}$ that produces this best classification probability corresponds the the best strong segmentation of the image.
It is this conditioning of the probability on a particular object class that provides the semantic content to direct the segmentation processs. The classification distance is then used as a quality metric for the segmentation that corresponds to that region combination. If the probabilities of membership, $P(\{X_k\}|C)$, for every class, *C* are too low, then the image is 'rejected', which implies the object of interest is not in the image.
The selection of these regions which will be combined into the final segmentation is analogous to feature selection in pattern recognition. There are a number of feature selection methods that can be adapted for region selection. The taxonomy for feature selection methods, shown in Fig. 6, divides these methods into three primary categories: (i) complete, (ii) heuristic, and (iii) random (Dash & Liu, 1997). These methods are the results of an extraordinary amount of research in the pattern recognition community, and are backed by both considerable empirical results as well as strong theoretical underpinnings. There is still no consensus as to which method is the best, since there is such a strong dependence of the performance of the algorithm on the data sets being analyzed (Dash & Liu, 1997). We have developed an approach in each of the major categories: an exhaustive search in the complete category, a Genetic Algorithm in the random category, and a forward sequential search in the heuristic category.

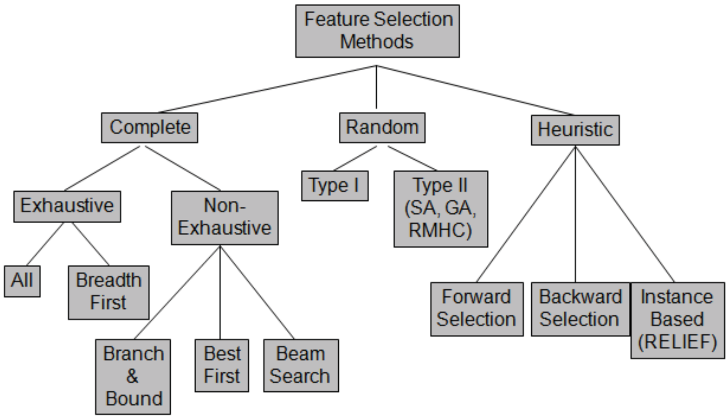


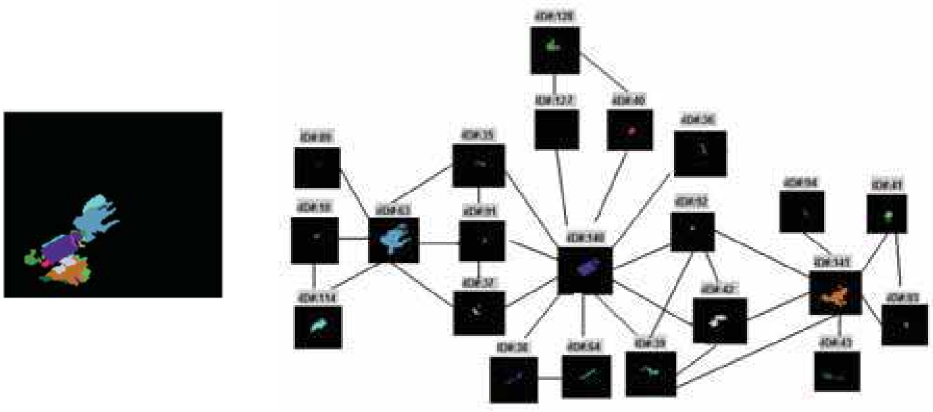
Fig. 6. Taxonomy of available feature selection methods.

The simplest algorithm is an exhaustive search through which all the possible region combinations are created and tested to find the best combination, which is feasible only when there are a limited number of regions in a cluster. For an exhaustive search, if there are N regions then the number of region combinations can rapidly become intractable since we must test the following number of combinations:

$$N_{combinations} = \sum_{k=1}^{N} \binom{N}{k}, \text{ where } \binom{N}{k} = \frac{n!}{k!(n-k)!}. \tag{7}$$

The summation over the number of regions is due to the fact that we do not know how many regions will be required to produce the best combination of blobs, and therefore every combination of every possible number of regions must be tested. For each of these possible number of regions, k, there are N *choose* k possible ways to select these regions from the complete set of N regions.

For all of the search methods, particularly for the exhaustive search, the number of possible region combinations can quickly become intractible, where for clusters consisting of as few as twenty regions a brute force search of every possible combination would require roughly one million combinations, and if the number of regions only increased modestly to twenty-five, the number of combinations would exceed 33 million. Fortunately, the total number of possible region combinations that must be explored is considerably less than this value which is actually an upper limit based on complete connectivity of all regions in the image. In reality the regions in an image are only *locally* connected which can easily be visualized using an adjacency graph. For region combiniing, only region combinations which satisfy an adjacency constraint must be tested. Fig. 7 (b) shows the adjacency graph for the cluster on Fig. 7 (a), and originally shown in Fig. 5. Here there are 22 regions in the cluster which for an exhasutive search of all possible combinations of all regions would result in 4.2 million combinations, however, the relatively sparse connectivity of the adjacency graph allows the sequential search algorithm to complete the analysis of this cluster with testing only 200 region combinations and correctly extracting the building of interest.



(a) (b)

Fig. 7. Graph representation of cluster from Fig. 5 (d): (a) image of cluster, and (b) graph representation of adjacency of cluster regions**.**

The adjacency of the regions creates a local connectivity in the graph, and it is this median local connectivity of the graph which determines the number of possible region combinations that must be explored. There are *n* posible selections for the first region to be selected, but then rather than *(n-1)* options for the second region, there are only *m* which is equal to the local connectivity of the first region. The number of possible regions for the third region then varies as *(m-1)*, etc. For the fourth region, the number of possible regions varies from *2*(m-1)* when the first three regions form a chain to *(m-2)* when they are all in a tight cluster adjacent to the first region. While the actual number of possible region selections cannot be calculated in closed form, the average number of region combinations which must be tested for each pass of the algorithm is:

$$\binom{N}{k}_{RAG} = n\left\langle \begin{array}{l} (m)(m-1)(m-1)\ldots(m-1), \\ (m)(m-1)(m-1)\ldots(m-k-1), \\ \ldots, \\ (m)(m-1)(m-2)\ldots(m-k) \end{array} \right\rangle, \tag{8}$$

Which is clearly significantly smaller than that estimated by *N-choose-k.*

Genetic Algorithm-based Region Combining

Genetic algorithms are a natural candidate for wrapper-based segmentation, since GAs can "successfully deal with combinatorial problems" (Kim, et al., 2000). Three key design issues must be addressed when using GAs: (i) the representation of the problem into a chromosome, (ii) the definition of a fitness function, and (iii) the the selection of cross-over and mutation strategies (Goldberg, 1989). Since we are using the GA for region selection, only the fact that a region is to be used in the segmentation must be encoded, which greatly simplifies the use of a GA for the wrapper-based segmenter versus other low-level segmentation approaches such as that described in (Bhandarkar & Zhang, 1999). The resultant encoding is a simple binary representation where the gene is set to one if the region is used, and set to zero if the region is not used.

The output from every iteration i of the GA-based region combiner algorithm is the set of regions for each member k of the population, $\{X_{ik}\}$ and the associated probability of correct classification for that member of the population, $P(\{X_{ik}\} \mid C)$ conditioned on the given pattern class, C.

The genetic algorithm uses this probability of correct classification as the fitness function for evaluating the population members for possible reproduction using a fitness-proportional selection scheme (Goldberg, 1989). The wrapper uses the fourth or second power of the probability of correct classification as the fitness proportionality measure, *Fitness(k)*, according to:

$$Fitness(k) = \frac{P(\{X_{ik}\} \mid C)^4}{\sum_{j=1}^{N} P(\{X_{ij}\} \mid C)^4} \text{ or } Fitness(k) = \frac{P(\{X_{ik}\} \mid C)^2}{\sum_{j=1}^{N} P(\{X_{ij}\} \mid C)^2}, \quad (9)$$

where N is the size of the population. The fourth power is employed if the variance of the distances is below a threshold, and the second power is employed as the population becomes more varied, to slightly reduce the dominating effects of a single highly fit parent. Not raising the classification distances to a higher power resulted in inadequate variation in the proportionality factors, thus leading to a nearly random selection scheme.

Pairs of parents are selected for mating using the roulette wheel selection mechanism, where each set of parents then has a probability $P_{crossover}$ of executing a cross-over to generate the children; otherwise the parents proceed intact, where we set $P_{crossover} = 0.85$. For the applications for which the wrapper has been employed, the region-labeling algorithm generates on average 20 regions, resulting in the chromosome having only 20 individual genes. Due to this relatively short sequence, a simple single-point cross-over scheme for the genetic operator has proven adequate. For each mating pair of parents chosen for cross-over, the cross-over point is randomly selected.

After the children are produced via the cross-over processing, the children experience mutation with a probability of any gene mutating being $P_{mutate} = 0.05$. Lastly, an elitist selection strategy is employed where the fittest 10% of the population prior to mating (this corresponds to the parents with highest fitness) are retained in the population (Back, 1996).

In order to ensure a diverse population of segmentation candidates is maintained two additional schemes are used to increase the diversity of the population. In the first scheme, if the variance of the fitness of the population falls below a threshold (i.e. the members of the population are becoming too similar), an additional mutation event is applied on the entire population, where this time $P_{mutate} = 0.25$. In the second scheme, if the fitness of the best member of the population has not improved in the last N iterations, where N=25, an additional mutation event is applied on the entire population with $P_{mutate} = 0.25$.

Sequential Search-based Region Combining

The sequential feature selection methods can be implemented in either a forward selection mode or a backward selection mode as can be seen from Fig. 6 under the heuristic methods. The forward selection mode begins with the empty set (an empty image) and then adds regions until the classification accuracy is maximized. The backward selection mode, on the other hand, begins with the complete image and removes regions until the classification accuracy is maximized. For the wrapper segmentation framework the forward selection is employed since the objects of interest are visually a fraction of the entire image. The forward selection algorithm that has been implemented is called the *plus-L-minus-R* algorithm, which has been identified as one of the more powerful heuristic methods for feature selection (Kudo & Sklansky, 2000). It begins with an initial set of regions, $\{X_0\}$ and then adds up to *L* regions per iteration and then after adding these *L* regions, tries region combinations where it subtracts up to *R* regions. The complete addition and then removal of regions is one iteration of the algorithm. The details of the algorithm are shown in Table 1. For the *plus-L-minus-R* implementation of the forward sequential search algorithm, the selection of *L* and *R* depends on the specific application and characteristics of the objects of interest within the images, for the airbag application where there were many regions that comprised the image we employed *L=5* and *R=3*, while for the tumor and the detection applications we employed *L*=*3* and *R*=*2*. The intitial number of regions to use is also an open parameter, which was five for the airbag application and two for the other two applications since the objects being processed were much smaller than the size of the image.

1)	For a given class *C*, create an initial set of regions $X_0 = \phi$, the empty set.
2)	Region Addition: At each stage *k* in the processing, test each region in the set of unselected regions and add region x_l if $P(\{X_k\} + x_l \mid C) \geq P(\{X_k\} \mid C)$, where $P(\{X_k\} \mid C)$ is the classification accuracy for the region set $\{X_k\}$, given class C. The output of this stage is a new subset of regions $\{X_{k+1}\} = \{\{X_k\}, x_{k+1}^{(1)}, x_{k+1}^{(2)} \ldots x_{k+1}^{(L)}\}$, where $x_{k+1}^{(i)}$ is the region with the ith best improvement in classification accuracy up to *L* regions.
3)	Region Removal: Test each region in the current selected region set, $\{X_{k+1}\}$, and remove each region x_r from the set if $P(\{X_{k+1}\} - x_r \mid C) \geq P(\{X_k\} \mid C)$, where $P(\{X_k\} \mid C)$ is the classification accuracy for the region set $\{X_k\}$ given class *C*. Continue testing and removing regions until all the regions in the current subset $\{X_k\}$ are tested, or until *R* regions have been removed.
4)	Record $P(\{X_k\} \mid C)$, and the corresponding subset of regions $\{X_k\}$, and return to step (2) unless the last region has been processed.

Table 1. *Plus-L- minus-R* forward sequential search algorithm for region combining.

3.2.3 Classification

Every possible combination of regions must be classified based on the class of interest to determine the goodness of the segmentation, however, prior to each classification, the features for the region combination must be computed. Recall to this point only the geometric moments have been employed to allow the features for each region combination to be quickly computed by adding or subtracting the moments for each region included in the combination. Recall from Equation (6), the geometric moments feature vector for a region combination is simply the sum of the feature vectors for every region that comprises the combination. This raw geometric moment vector is then converted to the desired invariant moments prior to classification. As a minimum the central moments must be used to make the object search translation invariant across the images and are computed by (Teague, 1980):

$$\mu_{mn} = \sum_{r=0}^{m} \sum_{s=0}^{n} \binom{j}{r}\binom{k}{s} (\bar{i})^{j-r} (\bar{j})^{k-s} M_{rs}, \tag{10}$$

where

$$\bar{i} = \sum_{j=1}^{Nrows} \sum_{i=1}^{Ncols} I(i,j) \cdot i \text{ and } \bar{j} = \sum_{j=1}^{Nrows} \sum_{i=1}^{Ncols} I(i,j) \cdot j \tag{11}$$

Depending on the desires of the user, the features of the object can also be made central scale invariant, rotation invariant, affine invariant and even projection (perspective) invariant (Suk & Flusser, 2004). More complex invariances, however, require more complex processing which impacts the throughput of the system. Also the more invariant the measures, the less discriminating the moments features can often be (Suk & Flusser, 2004). For the airbag suppression and the tumor applications, the sizes of the tumors were critical information so only central (translational invariant) moments were used. However, for the aerial surveillance application, the range to the objects of interest varied, and hence their size varied, which required central scale invariant moments. It is also important to note that not all objects require all invariances, for example when searching for bears, buildings, etc. as there is not a need to be fully rotationally invariant. Also the author has found that rotational invariance can be accomplished more cost effectively by adding a rotation generation function when creating the training database to create rotated examples of the training samples.

One key decision that must be made when employing moments is to decide the order of the moments being retained. The applications to be highlighted in Section 4 have varied from only fifth order for an aerial surviellance application designed to detect buildings to up to twenty-fifth order for detecting occupants in an automotive airbag application. The tumor application was in the middle of this range with tenth order.

While it is possible to use any of a number of possible classifiers in our wrapper method that provides a real-valued measure of the classification accuracy (or inversely classification distance), the k-nearest neighbor classifier has been used for the following reasons: (i) ease of implementation, (ii) non-parametric nature, (iii) demonstrated performance over a broad class of problems, and (iv) asymptotic convergence to the Bayes error rate (Jain et al., 2000) (Duda et al., 2000). The best results typically occur when the nearest neighbor classifier (*k=1)* is used. Additionally all of the moment values are used in the classification process (i.e. no feature selection is employed), since the complete set of features appears to be required to provide a good representation of the object shape. One last decision to be made for classification regards whether the features are normalized or not. The effects of normalization also varied with application, where we found that for the airbag suppression, un-normalized moments worked best, while the tumor and surveillance applications performed best when the features were normalized. This may be due to the fact that for the airbag suppression, the shapes were more complex and un-normalized features more fully captured and preserved the shape information that is providing the semantic information to the segmentation process.

4. Results

The wrapper framework has been demonstrated on three distinct applications, a vision system for automotive safety, an MRI analysis tool for automated breast cancer detection, and an aerial surveillance application. There has been considerable attention paid to developing 'smart' airbags that can determine not only if they should be deployed in a crash event, but also with what force they should be deployed. In May 2001 the U.S National Highway Transportation and Safety Administration (NHTSA) defined the Federal Motor Vehicle Safety Standard (FMVSS) 208 that mandated automatic airbag suppression when an infant is in the passenger seat. The detection of an infant in the seat defines a 2-class recognition problem where the classes are: (i) infant, and (ii) adults. An example of an input adult image, the preliminary segmentation after background removal and the resultant labeled image are provided in Fig. 8 (Farmer & Jain, 2005). For this particular application the background removal occured prior to low-level labeling since there was contextual information available regarding the knowledge of the empty vehicle which facilitated the removal of significant amounts of background information except the occupant and the seat. This system was tested using both the heuristic *plus-L-minus-R* algorithm and the Genetic Algorithm. For both algorithms, the central moments are converted to central-Legendre moment of order *(n+m)* for each region combination using (Teague, 1980):

$$L_{mn} = \frac{(2m+1)(2n+1)}{4} \sum_{l=0}^{m} \sum_{k=0}^{n} C_{ml} C_{nk} \mu_{lk}, \tag{12}$$

where C_{ml} are the coefficients defined by the Legendre polynomial generating function and μ_{lk} is the central moment of order *(l+k)*. Note for this application, the moments cannot be scale invariant since the size of the object is a critical factor in determining its class.

(a) (b) (c)

Fig. 8. Preliminary segmentation results for adult occupant, (a) adult image, (b) preliminary segmented adult, and (c) region labeled adult image (Farmer & Jain, 2005).

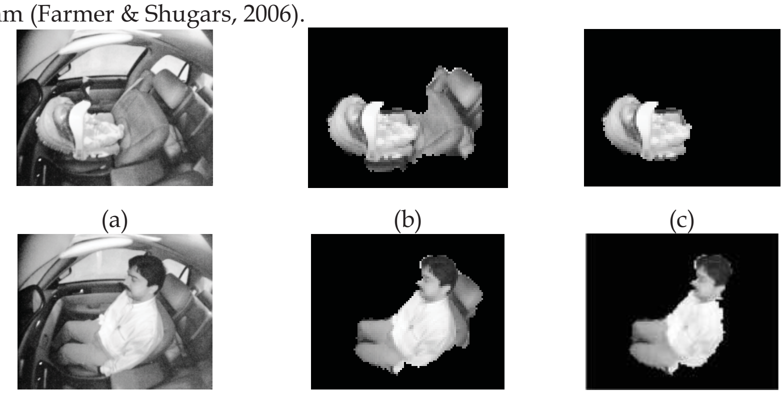
The classification results are provided in the confusion matrix in Table 2 for the *plus-L-minus-R* and in Table 3 for the genetic algorithm. The overall accuracy of the system provided by the *plus-L-minus-R* algorithm is $P_{correct}$(overall) is 91% with $P_{correct}$(infant) being 98.8% and $P_{correct}$(adult) being only 53.2% and with examples of correct segmentations shown in . Notice without shape information, accurate segmentation of these objects from the images would have been impossible since there is no low-level homogeniety constraint to differentiatethe object of interest from the background. Unfortunately, for the *plus-L-minus-R* algorithm the adult results are disappointing due to the high variability of the test images, which can be seen from an example incorrect segmentation in , where the occupant was moving forward and hence was not in the standard seating position. The *plus-L-minus-R* had trouble converging to the right answer on these conditions, but the testing of the GA on similar dataset improved the adult performance at a slight cost to the infant classification accuracy, as shown by its performance highlighted in Table 3. In summary $P_{correct}$(overall) is 88% with $P_{correct}$(infant) being 89.2% and $P_{correct}$(adult) being a much improved 83.4% (Farmer & Shugars, 2006). This performance improvement is due to the fact that the region selection space is a complicated search space with many local optima, and genetic algorithms have been shown to be more effective in these spaces (Kudo & Sklansky, 2000).

	True Infant	True Adult
Classified as Infant	1631	19
Classified as Adult	166	189

Table 2. Confusion matrix for the two-class suppression problem using *plus-L-minus-R* (Farmer & Jain, 2005).

	True Infant	True Adult
Classified as Infant	793	34
Classified as Adult	96	171

Table 3. Confusion matrix for the two-class suppression problem using a Genetic Algorithm (Farmer & Shugars, 2006).



(a) (b) (c)

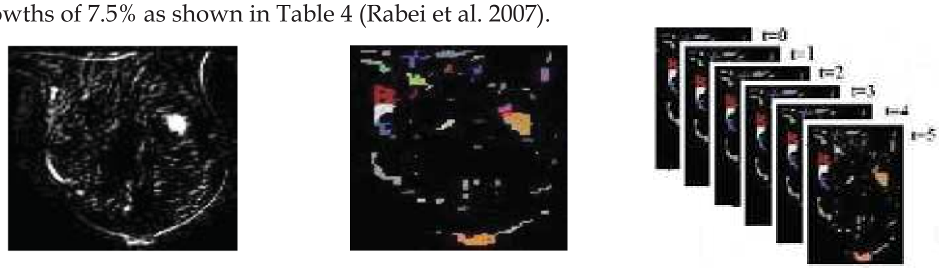
(d) (e) (f)

Fig. 9. Segmentation of an occupant images: (a) infant image, (b) preliminary infant segmentation, (c) final wrapper-based infant segmentation, (d) adult image, (e) preliminary adult segmentation, and (f) final wrapper-based adult segmentation (Farmer & Jain, 2005).

(a) (b) (c)

Fig. 10. Incorrect segmentation of an adult image: (a) adult image, (b) preliminary segmentation, and (c) final wrapper-based segmentation (Farmer & Jain, 2005).

The wrapper framework has also been applied to breast tumor detection (Rabei et al. 2007). Dynamic contrast-enhanced magnetic resonance imaging (DCE-MRI) has been identified as a valuable complementary technique for breast imaging. Unfortunately, while these multi-temporal image sequences provide new information, integrating and evaluating the much wider range of information is challenging task for human observers. The wrapper framework was used to direct the segmentation based on the underlying shape and temporal characteristics of the object of interest (Rabei et al. 2007). Examination of temporal kinetic patterns as measured for small regions of interest is a common method for characterizing lesion masses. These dynamic parameters cannot be computed for each pixel in every breast slice, due to processing complexity. Traditionally, these measures are computed by sampling pixels within a grid superimposed on the image, which can reduce sensitivity to detection of small tumors since much of the tissue within the grid cell is normal. The wrapper approach utilized the regions selected by the region combining and computed the dynamic parameters for each of these groupings, as can be seen in Fig. 11 (Rabei et al. 2007). These values are then used with the region shape information for tumor detection. The overall accuracy of the system is roughly 92% with the false positive diagnoses rate for normal patients as having either malignant or benign tumors of 4.5%, and a misdiagnosis rate for normal patients as either having malignant, benign, or suspicious growths of 7.5% as shown in Table 4 (Rabei et al. 2007).



(a) (b) (c)

Fig. 11. Intermediate breast tumor processing results, (a) input image, (b) labled image with background removed showing regions for combination, and (c) sequence of region combined images for dynamic analysis (Rabei et al. 2007).

Thus far, the wrapper has been demonstrated on a two-class problem for the airbag control system and a four-class problem for the tumor recognition system. The last application for which the wrapper framework has been applied is an aerial surveillance application addressing wide-area surveillance of disaster areas such as during hurricane Katrina, with an example wide-area image shown in Fig. 12. This is an object detection problem, which can be considered a single-class problem. The goal of the system is to detect manmade structures in wide-area imagery to ease the workload of image analysts who are searching for possibly stranded people in very remote rural areas. In this application, due to the immense sizes of the images, the first step in processing is a mosaicing process that divides the incoming image into a *4x4* grid, and each mosaic in the grid is then processed in parallel to reduce the processing time allowing it to benefit from multi-core architectures.

	True Normal	True Benign	True Suspicious	True Malignant
True Normal	925	23	31	21
True Benign	4	205	12	4
True Suspicious	7	19	343	6
True Malignant	3	2	4	91

Table 4. Results of wrapper framework applied to breast tumor detection (Rabei et al. 2007).

One other difference in processing is that the mode filtering step shown in Fig. 4 is bypassed since the objects tend to be relatively small in these massive images and the mode filtering distorted the shape characteristics of the objects of interest. The detection results for the wrapper on the image shown in Fig. 12 (b) are provided in Table 5, where the detection results are quite respectable. The quality of the segmentations and detections can be seen beginning with typical initial clusters and the resultant detections are provided in Fig. 13 and Fig. 14. These figures show the detected clusters in (a), the resultant combinations of regions that define the segmentations in (b), the region in the color image showing the object detection in (c), and the training sample that was used for the detection in (d).

(a) (b) (c)

Fig. 12. Surveillance images, (a) original wide-area image with buildings ,(b) zoom of highlighted region, and (c) image with no buildings.

	Combinations Detected as Objects	Combinations Detected as non-Objects
Actual Objects	5	0
Actual Clusters not Containing Object	2	7

Table 5. Wrapper Results on Image in Fig. 12(b).

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

Fig. 13. Best match for cluster: (a) Original blob cluster, (b) final blob combination, (c) image region and (d) best training sample.

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

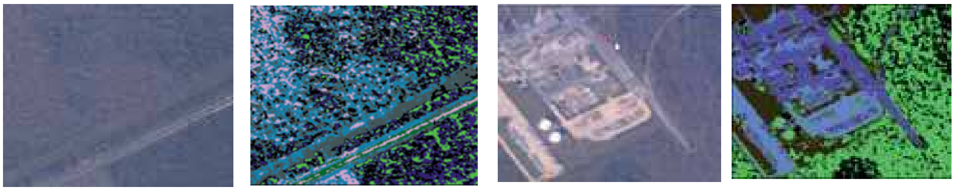
Fig. 14. Best match for cluster: (a) Original blob cluster, (b) final blob combination, (c) image region and (d) best training sample.

We also processed the entire image in Fig. 12 (a) and registered detections of buildings simply within each mosaic, so for each image there would be a total of sixteen possible detections. For analysis of this wide-area problem we quantify system performance in terms of recall and precision which are defined as:

$$\begin{aligned} \text{Recall} &= \left(\frac{\text{correct detections}}{\text{correct detections} + \text{missed detections}}\right) \\ \text{Precision} &= \left(\frac{\text{correct detections}}{\text{correct detections} + \text{false alarms}}\right) \end{aligned} \tag{13}$$

Unfortunately, the basic performance was not very impressive, with seven regions falsely having buildings detected, three with positive detections, and one missed detection, resulting in a Recall = .75 and Precision = 0.3.

There are two characteristics of the image segments where the wrapper framework had false detections. The first is where there are manmade entities such as parking lots and intersections of multiple roadways, which since the goal of the application is to detect manmade structures can only partially be considered false detections. The second cause of false detections occurs when the initial segmentation is severely over-segmented (we term this *hyper-segmentation*) which occurs when the image of interest has strong texture characteristics as shown in Fig. 15. For example, in these hyper-segmented regions there were on the order of 10^{100} to 10^{200} possible region combinations which are extraordinary. This high region count, and hence high number of region combinations explains the high false detection rate, since Borenstein and Mailk (Borenstein & Malik, 2006) pointed out that the segmented regions cannot be too small or else any object is possible to create. In this application, the hyper-segmentation can be avoided by implementing a texture and color-based low-level segmentation, which is beyond the scope of this paper. These conditions are easy to detect since they result in significant numbers of regions (typically over 400-600 where the normal number is less than 200). Thus the wrapper framework can also provide quality feedback regarding the initial segmentation, and redirect either the parameter selection or in this case the actual low-level feature set to use for labeling. When the hyper-segmented regions were removed from the calculation, the results are: Recall = 0.75, Precision = 0.6. This performance is more reasonable for a system that is designed to reduce operator workload.



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

Fig. 15. Problematic low-level segmentations, (a) hyper-segmentation due to significant texture from Mosaic (2, 4) from the Fig. 12 (a), showing texture, (b) the resultant color-based labelling using EM, (c) original image for showing under-segmentation, and (d) under-segmentation due to similarity in color of objects and the background roadway.

5. Conclusion

This chapter proposes an alternative paradigm for object segmentation that follows the wrapper methods of feature selection, where in this case the segmentation and the classification are *wrapped* together, and the classifier provides the metric for selecting the best segmentation. Rather than considering this method as yet another segmentation algorithm, the wrapper method is actually an alternative image segmentation framework, within which existing image segmentation algorithms may be executed. Unlike previous work in image segmentation, the proposed system makes no assumptions regarding the homogeneity of the object of interest. It attempts to bridge the semantic gap in image segmentation by considering the shape of the desired object, rather than relying on lower level features such as color or texture. The approach has been implemented with two different region selection algorithms, the heuristic *Plus-L Minus-R* algorithm and a genetic algorithm, while for small region combinations an exhaustive search is applied. The wrapper framework has been demonstrated on three very different applications, a vision-based automotive occupant sensing system, a breast tumor recognition system using MRI, and an aerial surveillance application for disaster assessment. In all cases, the resultant segmentations were often of high quality and would have been impossible without the semantic information provided by the shape of the object of interest. In the surveillance application, the results were more dependent on the low-level segmentation, caused by hyper-segmentation due to high texture images. Future work will address integrating more powerful low-level segmenters other than the EM algorithm. Current research work is directed at developing a complete content-based image query system using the wrapper framework to support the search through an image database of user defined shapes of interest. Shape-based Content-based Image Retrieval (CBIR) is currently a very active area of research and the wrapper framework may provide an effective means for integrating shape information into the search process.

6. References

Athanasiadis, T., Mylonas, P. Avrithis, Y., & Kollias, A. (2007). “Semantic image segmentation and image labelling”, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 3, 2007.

Back, T. (1996). *Evolutionary Algorithms in Theory and Practice*, Oxford Press.

Belongie, S.; Carson, C.; Greenspan, H. & Malik, J. (1997). "Color and texture-based image segmentation using EM and its application to content-based image retrieval," *Proc. International Conference on Computer Vision (ICCV)*, pp. 675-683, 1997.

Bhandarkar, S. M. & Zhang, H. (1999). “Image segmentation using evolutionary computation”, *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 1, 1999.

Bhanu, B.; Lee, S. & Das, S. (1995). “Adaptive image segmentation using genetic and hybrid search methods”, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 4, pp. 1268-1290, Oct. 1995.

Bhanu, B. & Fonder, S. (2000). “Learning-based interactive image segmentation”, *Proc. of IEEE International Conference on Pattern Recognition*, pp. 1299-1302, 2000.

Bhanu, B. & Peng, J. (2000). “Adaptive integrated image recognition and segmentation”, *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, vol. 30, no. 4, pp. 427-441, Nov. 2000.

Borenstein, E. & Ullman, S. (2002). “Class-Specific, Top-Down Segmentation”, In: *Lecture Notes in Computer Science*, vol. 2351, Springer-Verlag, pp. 109-122, 2002.

Borenstein, E. & Malik, J. (2006), “Shape Guided Object Segmentation”, *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 969 – 976, 2006.

Bouman, C.A. & Shapiro, M. (1994). “A multi-scale random field model for bayesian image segmentation,” *IEEE Transactions on Image Processing*, vol. 3, no. 2, pp. 162-177, 1994.

Cour, T. & Shi, J. (2007). “Recognizing objects by piecing together the segmentation puzzle”, *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1-8, 2007.

Dash, M. & Liu, H. (1997). “Feature selection for classification”, *Intelligent Data Analysis*, vol. 1, pp. 131-156, 1997.

Datta, R.; Joshi, D.; Limm, J. & Wang, J.Z. (2008). “Image retrieval: Ideas, influences, and trends of the new age”, *ACM Computing Surveys,* vol. 40, no. 2, pp. 5:1-5:60, 2008.

Duda, R.O.; Hart, P.E. & Stork, D.G. (2000). *Pattern Classification*, 2nd edition, Wiley, New York, 2000.

Duin, R.P.W. (1996). “A note on comparing classifiers,” *Pattern Recognition Letters*, vol. 17, no. 5, pp. 529-536, 1996.

Falconer, K. (2004). *Fractal Geometry Mathematical Foundations and Applications*, 2nd edition, Wiley, New York, 2004.

Farmer, M. & Jain, A. (2005). “Wrapped based approach to image segmentation and classification”, *IEEE Transactions in Image Processing*, vol. 14 no.12, pp. 2060-2072, 2005.

Farmer, M. & Shugars, D. (2006). “Application of Genetic Algorithms for Wrapper-based Image Segmentation and Classification”, *Proc. of IEEE World Congress on Evolutionary Computation,* pp. XXX, 2006.

Farmer, M. (2008). “Application of the wrapper framework for object detection”, *Proc. of the IEEE International Conference on Pattern Recognition*, pp. XXX, Tampa, Florida, 2008.

Flusser, J. & Suk, T. (1999). “On the calculation of image moments”, *Research Report #1946, Institute for Information Theory and Automation, Academy of Sciences of the Czech Republic*, 1999.

Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

Jain, A.K., Duin, R.P.W. & Mao, J. (2000). “Statistical pattern recognition: A review”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4-37, 2000.

Jing, F.; Li, M.; Zhang, H.-J. & Zhang, B. (2004). “An efficient and effective region-based image retrieval framework”, *IEEE Transactions on Image Processing*, vol. 13, no. 5, 2004.

Kim, E.Y.; Park, S.H. & Kim, H.J. (2000). “A genetic algorithm-based segmentation of Markov random field modeled images”, *IEEE Signal Processing Letters*, vol. 7, no. 11, 2000.

Ko, B. & Byun, H. (2005). “FRIP: A region-based image retrieval tool using automatic segmentation and stepwise Boolean AND matching”, *IEEE Transactions on Multimedia*, vol. 7, no. 1, 2005.

Kohavi, R. & John, G.H. (1998). “The wrapper approach”, In: *Feature Extraction, Construction and Selection: A Data Mining Perspective*, Kluwer Academic, pp. 33-50, 1998.

Kudo, M. & Sklansky, J. (2000). “Comparison of algorithms that select features for pattern classifiers”, *Pattern Recognition*, vol. 33, pp.25-41, 2000.

Lee, M.W. & Cohen, I. (2004). “Proposal maps driven MCMC for estimating human body pose in static images” *Proc. IEEE Conference on Computer Vision and Pattern Recognition,* pp.334-341, 2004.

Li, J.; Wang, J.Z. & Wiederhold, G. (2000). “IRM: Integrated region matching for image retrieval”, *Proc. 8th ACM Intl Conf. on Multimedia*, pp. 147-156, 2000.

Luo, J. & Guo, C. (2003). “Perceptual grouping of segmented regions in color images”, *Pattern Recognition*, vol. 36, pp. 2781-2792, 2003.

National Oceanic and Atmospheric Administration website, http://ngs.woc.noaa.gov/katrina/KATRINA0000.HTM.

Pal, N. R. & Pal, S. K. (1993). "A review on image segmentation techniques", *Pattern Recognition*, vol. 26, no. 9, pp. 1277-1294, 1993.

Rabiei, H.; Mahloojifar, A. & Farmer, M. (2007). “Providing context for tumor recognition using the wrapper framework”, *IEEE International Symposium on Biomedical Imaging*, 2007.

Safar, M.; Shababi, C. & Sun, X.(2000). “Image retrieval by shape: A comparative study”, *Proc. IEEE International Conference on Multimedia and Exposition*, pp. 141-144, 2000.

Shi, J. & Malik, J. (2000). “Normalized cuts and image segmentation”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905, Aug. 2000.

Sifakis, E., Garcia, C. & Tziritas , G. (2002). “Bayesian Level Sets for Image Segmentation”, *Journal of Visual Communication and Image Representation*, vol. 12, no. 1/2, pp. 44-64, 2002.

Smeulders, A.W.M.; Worring, A.; Santini, S.; Gupta, A. & Jain, R. (2000). “Content-based image retrieval at the end of the early years”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349-1380, Mar. 2000.

Spiliotis, I.M. & Mertzios, B.G. (1998). "Real-time computation of two-dimensional moments on binary images using image block representation", *IEEE Transactions Image Processing*, vol. 7, no. 11, pp. 1609-1615, 1998.

Suk, T. & Flusser, J. (2004). "Projective moment invariants", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 10, 2004.

Teague, M.R. (1980). "Image analysis via the general theory of moments", *J. Opt. Soc. Amer*. vol. 70, no. 8, pp. 920-930, 1980.

Tu, Z. & Zhu, S.-C. (2002). "Image segmentation by data-driven Markov chain monte carlo" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 657-673, May 2002.

Tu, Z.; Chen, X.; Yuille, A.L. & Zhu, S.-C. (2003). "Image parsing: Unifying segmentation, detection, and recognition", *Proc. IEEE International Conference on Computer Vision*, pp. 18-25, 2003.

Veltkamp, R.C. & Hagedorn, M. (2001). "State-of-the-art in shape matching", In *Principles of Visual Information Retrieval*. pp. 87-119, Springer, 2001.

Yang, J. & Honavar, V. (1998). "Feature subset selection using a genetic algorithm", *IEEE Intelligent Systems*, March-April, pp. 44-49, 1998.

Yoshitake, A. & Ichikawa, T. (1998). "A survey on content-based retrieval for multimedia databases", *IEEE Transactions on Knowledge and Data Engineering*, vol. 11, no. 1, pp. 81-93, 1999.

Yu, S.X.; Gross, R. & Shi, J. (2002). "Concurrent object recognition and segmentation by graph partitioning", In: *Advances in Neural Information Processing Systems 15*, 2002.

Pattern Recognition

Edited by Peng-Yeng Yin

ISBN 978-953-307-014-8

Hard cover, 568 pages

Publisher InTech

Published online 01, October, 2009

Published in print edition October, 2009

For more than 40 years, pattern recognition approaches are continuingly improving and have been used in an increasing number of areas with great success. This book discloses recent advances and new ideas in approaches and applications for pattern recognition. The 30 chapters selected in this book cover the major topics in pattern recognition. These chapters propose state-of-the-art approaches and cutting-edge research results. I could not thank enough to the contributions of the authors. This book would not have been possible without their support.

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821