Chapter 4

Segmentation

Segmentation is the low-level operation concerned with partitioning images by determining disjoint and homogeneous regions or, equivalently, by finding edges or boundaries. The homogeneous regions, or the edges, are supposed to correspond to actual objects, or parts of them, within the images. Thus, in a large number of applications in image processing and computer vision, segmentation plays a fundamental role as the first step before applying to images higher-level operations such as recognition, semantic interpretation, and representation. Until very recently, attention has been focused on segmentation of gray-level images since these have been the only kind of visual information that acquisition devices were able to take and computer resources to handle. Nowadays, color imagery has definitely supplanted monochromatic information and computation power is no longer a limitation in processing large volumes of data. The attention has accordingly been focused in recent years on algorithms for segmentation of color images and various techniques, often borrowed from the background of gray-level image segmentation, have been proposed. This paper provides a review of methods advanced in the past few years for segmentation of color images.

The desirable characteristics that a good image segmentation should exhibit with reference to gray-level images were clearly stated in [RMH85]: "Regions of an image segmentation should be uniform and homogeneous with respect to some characteristics such as gray tone or texture. Region interiors should be simple and without many small holes. Adjacent regions of a segmentation should have significantly different values with respect to the characteristic on which they are uniform. Boundaries of each segment should be simple, not ragged, and must be spatially accurate."
A more formal definition of segmentation, accounting for the principal requirements listed above, can be given in the following way [KSF81, NRP93, RJR95]: Let \( I \) denote an image and let \( H \) define a certain homogeneity predicate; then the segmentation of \( I \) is a partition \( P \) of \( I \) into a set of \( N \) regions \( R_n \), \( n = 1; \ldots; N \), such that:

1) \( \bigcup_{n=1}^{N} R_n = I \) with \( R_n \cap R_m \neq \emptyset \), \( n \neq m \);
2) \( H(R_n) = \text{true} \) for all \( n \); and
3) \( H(R_n \cup R_m) = \text{false} \) for all adjacent \( R_n \) and \( R_m \).

Condition 1) states that the partition has to cover the whole image; condition 2) states that each region has to be homogeneous with respect to the predicate \( H \); and condition 3) states that the two adjacent regions cannot be merged into a single region that satisfies the predicate \( H \).

Segmentation is an extremely important operation in several applications of image processing and computer vision, since it represents the very first step of low-level processing of imagery. As mentioned above, the essential goal of segmentation is to decompose an image into parts which should be meaningful for certain applications [RMH85]. For instance, in digital libraries large collections of images and videos need to be catalogued, ordered, and stored in order to efficiently browse and retrieve visual information [SID96, SIS98]. Color and texture are the two most important low-level attributes used for content based retrieval of information in images and videos. Because of the complexity of the problem, segmentation with respect to both color and texture is often used for indexing and managing the data [MJS91].

In [LLS99, LLS01], an extensive bibliography is reviewed dealing with more recent color image segmentation approaches from which the most interesting ones are succinctly described next. In these surveys, the outstanding feature defining a segmentation is that the decomposition of an image into regions should be significant in relation to the application that is using such results.

These works classify segmentation techniques into three main categories:

1. Feature-space based techniques.
   - Clustering.
   - Adaptive k-means clustering.
   - Histogram thresholding.

2. Image-domain based techniques.
   - Split-and-Merge.
   - Region-growing.
   - Graph-theoretical techniques.
CHAPTER 4. SEGMENTATION

- Edge-based techniques.
- Neural networks.


4.1 Feature-Space Based Techniques

Color is a constant property of the surface of each object within an image and each pixel of the color image can be mapped into a certain color space. Thus different objects present in the image will manifest themselves as clusters or clouds of points. The spreading of these points within each cluster is mainly determined by color variations due to shading effects and to the noise of the acquisition device. On the other hand, if instead of mapping pixels into color spaces, we build some ad hoc histograms upon color features, such as hue, for instance, it is likely that the objects will appear as peaks within these histograms.

Therefore, the problem of segmenting the objects of an image can be viewed as that of finding some clusters, according to the first strategy mentioned above, or as that of finding the peaks of some opportune histograms, according to the second strategy. These two approaches work within a certain feature space, which may be one of the color spaces and they generally neglect the spatial relationship among colors.

4.1.1 Clustering Techniques

Clustering can be broadly defined as a non supervised classification of objects in which one has to generate classes or partitions without any a priori knowledge. Analogous to the definition of segmentation given before in [NPP93], the problem of clustering can be precisely stated as, once given a certain number of patterns, determining the set of regions such that every pattern belongs to one of these regions and never to two adjacent regions at the same time. Classification of patterns into classes follows the general common sense principle that objects within a class should show a high degree of similarity while not across different classes, where they should exhibit very low affinity.

One among the commonest algorithms that have been proposed in the literature of cluster analysis is the k-mean clustering [SHP98], widely adopted in vector quantization and data compression. A fuzzy version of this is commonly used in a number of works referred to in [LLS99, LLS01], as well as the closely related approach of probabilistic clustering. A comparison between crisp and
CHAPTER 4. SEGMENTATION

fuzzy versions of that algorithm can be found in [SRR95]. ISODATA is another algorithm often used for color space clustering [KTK99].

Another interesting and fruitful approach is the mean-shift algorithm reported in [DCM97, DCP99, DCP02]. Similarly to the problem of finding function extremes by gradient minimization, color clusters are found in this approach by computing the position in the feature space where the mean value within an image region shows the minimum variation in respect to other neighboring positions. Recently, mean-shift has also been extended to cope with the issue of tracking objects [DCP03].

Competitive learning based on the least-square criterion is employed in [TUM94], whereas the theory of connected components is adopted in [WWC97]. An original technique proposed in [NHC98] adopts the constrained gravitational clustering. Two points within a color space are modeled as two massive particles having an interaction according to the Newton's gravitational law. The net force on each particle determines the collapse of points into clusters whose number is governed by a given force-effective function. Yet, in [KUC94] color space is represented by way of a tree and clustering is achieved by simplification of that tree. This approach is an open door to the introduction of the closely related approach of graph theory, which will be separately reviewed due to its relevance.

Despite it is not included in any of the reviews referred to here, it is our strong believe that another important clustering strategy which recently has gained momentum is the probabilistic model-based approach to unsupervised learning that uses finite mixtures for the statistical modeling of data [AKJ88, AKJ00, GMD00]. Finite mixtures naturally model observations which are assumed to have been produced by one of a set of alternative unknown sources selected at random. Inferring the parameters of these sources and identifying which source produced each observation lead to a clustering of the set of observations. With this model-based (parametric) approach for clustering, opposed to heuristic methods like the aforementioned k-means of hierarchical growing methods [AKJ88], issues like the selection of the number of clusters or the assessments of the validity of a given model can be addressed in a more formal way.

The standard method used to fit finite mixture models is the Expectation-Maximization (EM) algorithm [DLR77, GMT97, GMD00] which converges to a Maximum Likelihood (ML) estimate of the mixture parameters. However, EM for finite mixture fitting is known to have several
drawbacks, namely, it is local (greedy), sensitive to initialization and, for a certain type of mixtures, it may converge to the boundary of the parameter space leading to meaningless estimates, apart from the issue of selecting the number of components.

Among the pile of versions and heuristics used to implement the EM algorithm, the work in [MAF02] is outstanding for simultaneously dealing with all the problems mentioned before. An inference criterion is proposed that automatically selects the number of components, greatly unsensitizes EM to initialization, and avoids the finicking problem of reaching the boundaries of the parameter space. More recently, the same authors in [MHC04] propose the concept of feature saliency and introduce an EM algorithm that estimates it as a mixture-based clustering.

Other color image segmentation approaches that use EM are the early work in TYI98] and those of [BCGM98, CBGM02], where the EM process is driven both in color and texture, and is extensively applied to retrieve images from large and varied collections by means of their content. Finally, in [CPP00] instead of using the local iterative scheme, a deterministic annealing EM is proposed to provide a global optimal solution for the ML parameter estimation.

4.1.2 Adaptive k-means Clustering Techniques

A special classification has to be devoted to a class of segmentation algorithms that combines the idea of k-means clustering with the properties of local adaptivity to color regions and of spatial continuity. The aforementioned clustering techniques assign pixels to clusters only on the basis of their color and no further spatial constraints are imposed.

In order to include spatial constraints the work in [TNP92] proposes a generalization of the k-means clustering algorithm which considers the segmentation of gray-level images as a Maximum A Posteriori (MAP) probability estimation problem. The extension of this technique to color images is proposed in [MMC94]. The estimated segmentation is defined as the one that maximizes the posterior probability of the segmentation provided the observed data in the image.

By using Bayes's rule, it is the minimum of the product between the image prior and the conditional probability of the image given a certain segmentation. A Gibbs Random Field (GRF) is used in [SGD84, SZL95] as an image prior to model and enforce spatial homogeneity constraints. Conditional probability is modeled as a multivariate Gaussian distribution with a space-varying mean function.
CHAPTER 4. SEGMENTATION

The algorithm alternates from MAP estimation to local determination of class means, which are initially constant for each region and equal to k-means cluster centers. Interactively, the algorithm then updates those means by averaging them over a sliding window whose size progressively decreases, starting with global estimates and progressively adapting them to the local characteristics of each region.

This algorithm has been further extended in [ESA96], where color image segmentation and edge linking are combined, applying a split & merge strategy to enforce edge consistency. Besides, in [JLR97, JLR98] the algorithm described in [MMC94] is modified to accept in the former a new color space and metric, which is claimed to provide physically more coherent segmentations, and derivative priors combining both region-based and edge-based statistics in the latter.

4.1.3 Histogram Thresholding Techniques

Histogram thresholding is among the most popular techniques for segmenting gray-level images and several strategies have been proposed [KSF81, RMH85, NPP93]. In fact, peaks and valleys in one-dimensional histograms can be easily identified as objects and backgrounds in gray-level images. In the case of color images, things are a little bit more complex since one has to identify different parts of a scene by combining peaks and valleys in three histograms or by partitioning a whole 3D histogram. A common problem with histogram is that noise often gives rise to spurious peaks and thus to segmentation ambiguities. To prevent this, some smoothing provisions are usually adopted.

Usually, pixel color is distributed into three histograms which are independently restricted by thresholds, e.g., by maximizing the within-group variance and combining the three results with a predicate logic function afterwards [MCM98]. In [LSM98] a watershed scheme is adopted to segment either 2D (chromaticity) or 3D histogram from a color image. Histograms are coarsened through convolution with a spherical window to avoid over segmentation. In [DCT95] only hue information is exploited and, therefore, it is suggested a circular histogram thresholding since hue is an angular attribute. Histogram smoothing is achieved by means of a scale-space filter. Other works use the whole HSI color space despite segmentation is undergone through only one coordinate, either hue or intensity. In [CSA97] a fast segmentation algorithm is suggested which resorts to a pre-clustered chromaticity plane after quantization of the HSV space represented into orthogonal Cartesian coordinates.
The work in [KSI96] singles out faces from color images by defining appropriate domains corresponding to skin-like regions within the HSV space. Robustness against changes in illumination and shadows is obtained by disregarding the luminance (V). In [GYM98] an entropy-based thresholding which assumes that patterns in the feature space are generated by two distinct sources, called modes and valleys. First, patterns are classified in either categories by using entropy thresholding and then the number of modes in the feature space is computed employing a modified Akaike's information criterion.

An alternative way of smoothing histograms and achieving better segmentations is by means of fitting a family of curves or density functions to shape observations. Thus, the distribution of the chrominance of the objects in a scene is modeled in [ESA95] as a Gaussian PDF allowing this way an adaptive setting of object-class thresholds. In [LJL94] an adaptive threshold function for both RGB and HSI spaces is devised by using B-splines. Another manner of smoothing hue histograms is suggested in [LLS98] by working with the low-low band of the wavelet transform of the image to be segmented.

4.2. Image-Domain Based Techniques

Almost all the segmentation algorithms of the previous section exclusively operate in some feature spaces. Thus, the regions (segments) they return are expected to be homogeneous with respect to the characteristics represented in these spaces; however, there is no guarantee at all that these regions also show spatial compactness, which is a second desirable property in segmentation applications beside homogeneity. In fact, cluster analysis and histogram thresholding account in no way for the spatial locations of pixels; the description they provide is global and it does not exploit the important fact that points of a same object are usually spatially close due to surface coherence [RJR95]. On the other hand, if pixels are clustered exclusively on the basis of their spatial relationships, the end result is likely to be with regions spatially well connected but with no guarantee that these regions are also homogeneous in a certain feature space.

In the literature of segmentation of gray-level images, a great many techniques have been suggested that try to satisfy both feature-space homogeneity and spatial compactness at the same time [RMH85, KSF81]. The latter is ensured either by subdividing and merging or by progressively growing image regions, while the former is adopted as a criterion to direct these
two processes [RMH85, KSF81, ARA82, RJR95]. According to the strategy preferred for spatial
grouping, these algorithms are usually divided into split-and-merge and region growing
techniques; this distinction may also be extended to the corresponding algorithms for color image
segmentation which will be analyzed in the following sections.

4.2.1 Split-and-merge Technique
A common characteristic of these methods is that they start with an initial inhomogeneous
partition of the image (usually the initial segment is the image itself) and they keep performing
splitting until homogeneous partitions are obtained. A common data structure used to implement
this procedure is the quadtree representation [RJR95, STB92] which is a multiresolution scheme.
After the splitting phase, there usually exist many small and fragmented regions which have to be
somehow connected. The merging phase accomplishes this task by associating neighboring
regions and guaranteeing that homogeneity requirements are met until maximally connected
segments can be produced. The region adjacency graph (RAG) is the data structure commonly
adopted in the merging phase [RJR95, STB92]. In many algorithms, smoothness and continuity
of color regions are enforced with the adoption of a Markov Random Field (MRF) [GRC83,
SGD84] which basically is a stochastic process characterized by the following property: the
conditional probability of a particular pixel taking in a certain value is only a function of the
neighboring pixels, not of the entire image. Besides, the Hammersley Clifford theorem establishes
the equivalence between MRF's and Gibbs distributions [SGD84].

Panjwani and Healey [DKP95] model color texture in RGB components by means of a Gaussian
Markov Random Field (GMRF) which embeds the spatial interaction within each of the three
color planes as well as the interaction between different color planes. In the splitting phase, the
image is recursively partitioned into square regions until each of them contains a single texture
described by a color GMRF model. This phase is followed by an agglomerative clustering phase
which consists of a conservative merging and of a stepwise optimal merging process.

Liu and Yang [JLY94] define instead an MRF on the quadtree structure representing a color
image and use the above mentioned equivalence with a Gibbs distribution. With a relaxation
process [ARA82] they control both splitting and merging of blocks in order to minimize the
energy in the Gibbs distribution; this is shown to converge to a MAP estimate of the
segmentation.
Numerous variations in the split-and-merge strategies have been investigated. In [MCH97] a k-means algorithm is used for both classifying the pixels in the splitting phase and grouping pattern classes in the merging phase. In [VAC97] the splitting is initially performed by segmenting the luminance and then refined by checking the chrominance homogeneity of the obtained regions; the merging is based on an ad hoc cost function. In [SJH98] the splitting is operated with the watershed transform [HDC77] of the gradient image of the luminance component simplified by a morphological gray-scale opening [RJR95, AKJ89, RCG92]; the merging step is realized with a Kohonen's self-organizing map (SOM) [STB92]. Shafarenko et al. [LSM97] apply instead the watershed transform to the L*u*v* gradient of images and merge the patches of the watershed mosaic according to their color contrast until a termination criterion is met. A similar splitting approach is adopted in [KSC94] whereas the merging phase is performed by iteratively processing the RAG constructed upon the resulting over segmented regions. Also Round et al. [AJR97] employ a split-and-merge strategy for segmentation of skin cancers; the splitting phase is based on a quad-tree representation of the image and the following conservative merging is performed with a RAG.

Gevers et al. [TGV94, TGA97] believe that split-and-merge algorithms based on a quadtree structure are not able to adjust their tessellation to the underlying structure of the image data because of the rigid rectilinear nature of the quadtree structure; therefore, they suggest replacing it with an incremental Delaunay triangulation [STB92]. A further alternative possibility is to use Voronoi diagrams [STB92] as proposed by Schettini et al. [RSM94] and by Itoh and Matsuda [SII96].

Broadly speaking, we can fit within the class of split-and-merge techniques also some algorithms based upon differential equations and pyramidal data structures. At the first glance, they do not appear to belong to this category since the strategies they adopt to achieve segmentation are rather different from those reviewed so far; but a more careful look into them will bring to light an underlying split-and-merge idea.

The usefulness of pyramidal representation of images for segmentation was pointed out by Burt et al. [PJB81] about two decades ago and ever since a number of methods to segment images by working with pyramids have appeared. It is well-known that pyramids are data structures in which images can be represented at different resolutions (fine-to-coarse) by means of tapering layers recursively obtained by averaging and down sampling their respective underlying layers.
CHAPTER 4. SEGMENTATION

[RJR95] (the finest layer at the bottom of a pyramid is the image itself). Thus, father-son relationships can be naturally introduced between adjacent layers of pyramids; segmentation can be achieved with a pyramid-linking process [PJB81] based on a tree data structure where the values of the fathers at a certain high layer are propagated down to the sons of the lowest level. The construction of a pyramid can be regarded as a splitting phase while the subsequent linking process can be seen as a merging phase. Recently, Lozano and Laget [VLB96] have suggested fractional pyramids for segmentation of color images and Ziliani and Jensen [FZB98] have proposed a modified version of the linking approach of [PJB81].

4.2.2 Region Growing Techniques
An homogeneous region of an image may be obtained through a growth process which, starting from a pre-selected seed, progressively agglomerates points around it satisfying a certain homogeneity criterion; the growth process stops when no more points can be added to the region. The region growing techniques are mainly aimed at processing single regions; nevertheless, by combining different and subsequent growth processes, one may agglomerate in regions all the points of an image, obtaining this way its segmentation. After a region growing procedure, there might exist some very small regions or there could be two or more neighboring regions grown at different times exhibiting similar attributes. A common post-processing provision consists therefore in a merging phase that eliminates such instances by generating broader regions.

The region growing can be considered a sequential clustering or classification process [ARA82]; thus the dependence of the results on the order according to which the image points are processed has to be accounted for. The main advantage offered by this kind of techniques is that the regions obtained are certainly spatially connected and rather compact. As for the clustering techniques of Section 4.1.1, where a similar problem arises in the feature space, also for the region growing techniques one is faced with the problem of choosing suitable seed points and an adequate homogeneity criterion.

Tremeau and Borel [ATN97] suggest several different homogeneity criteria operating in RGB coordinates. In a first phase, they generate a certain number of connected regions with a growing process and, in a second phase, they merge all the regions having similar color distributions; after the second phase, the regions have therefore homogeneous colors but they may be disconnected. Kanai [YKI98] develops a segmentation algorithm which resorts to both color and intensity information. The markers (seeds) are extracted from intensity via morphological open-close
CHAPTER 4. SEGMENTATION

operations and from color through quantization of the HSV space; joint markers are defined as
the sets comprising both kinds of markers. A region growing process based on a watershed
algorithm starts from these joint markers. A region merging process eventually reduces the
number of segmented regions.

In [BCM97], the initial seeds are generated by retaining the significant local minima of the
magnitude of the color image gradient; however, with this algorithm the two following situations
might arise: 1) there is more than one seed per region; 2) small objects do not have any seed. The
authors devise a procedure for obtaining markers having a one-to-one correspondence with the
image regions. The region growing is performed with a watershed-like algorithm proposed by the
authors and working on the original color image instead of on a gradient image.

Deng et al. [YDB99] determine a limited number of color classes within an image through color
quantization and propose a criterion for “good” segmentation based on them. The application of
this criterion within local windows and at multiple scales generates J-images in which high and
low values respectively correspond to possible region boundaries and to region centers. A region
growing method is adopted where the seeds are the valleys of the J-images; the resulting over
segmentation is finally removed with a merging phase.

Rehrmann and Priese [VRL98] suggest using a special hexagonal topology in a hierarchical
region growing algorithm which results independent of the starting point and of the order of
processing. Ikonomakis et al. [NIK98] develop an algorithm to segment both gray-scale and
videophone-type color images; the procedure is a standard region growing process followed by
region merging. Color homogeneity is tested with measurements in the HSI space.

If one define a cluster as a collection of touching pixels that have almost the same color while the
change in color is gradual," the fuzzy nature of the segmentation problem can be emphasized.
Moghaddamzadeh and Bourbakis [AMN94, AMN97] have adopted this outlook of the problem
and advanced two algorithms working in RGB coordinates to implement a region growing
strategy for both fine and coarse segmentation of color images. A fuzzy approach for region
growing segmentation is adopted also in [TCP96] whose algorithm is based upon several
linguistic rules defining relationships among hue, chroma, and intensity. Colantoni and Laget
[PBC97] compare the results of four different algorithms obtained by the various combinations of
region growing and watershed transform in a pre-segmentation step and in the actual
segmentation algorithm. Images are represented in L’ab\* coordinates and handled by means of RAG’s and contour graphs.

4.2.3 Graph Theoretical Techniques
Another interesting approach is the one based on graph theory. The goal here is to partition a graph describing the whole image into a set of connected components that correspond to image regions. There are at least two ways of doing so. On the one hand, there are splitting methods that partition a graph by removing superuous edges. On the other hand, region-growing methods join components as a function of the attributes of nodes and edges. Next, some of these graph-partitioning approaches are briefly described.

The most efficient graph-based algorithms use fixed thresholds and purely local measures to find regions. For example, the approach in [CTZ71] is based on breaking large edges in a Minimum Spanning Tree (MST) of the graph. A more recent method [ZWR93] is based on the computation of the minimum cut in the graph representing an image. The cut criterion is designed to minimize the similarity between regions that are being split. This approach captures non local properties of the image but requires more than nearly linear time, in contrast with the more efficient methods described bellow that are based upon local information. Other refinements of such methods can be found in [JSJ97, JSS98], where a normalized version of the minimum cut is computed. For a wider review on this sort of approaches, refer to the works in [UEG97, POF98].

In [RUG97] a measure of local variability is employed to decide which edges to remove. Local measures just rely upon the nearest neighbors of points and are not enough to get a reasonable glimpse of the whole image variability since they do not capture non local properties. This issue is specifically tackled in [PFF98]. Another graph-theoretical work in [JPW98] presents computationally “inexpensive” algorithms for probability simulation and simulated annealing, such as Hastings's and generalized Metropolis's algorithms. To reduce the computational burden, a hierarchical approximation is proposed minimizing at each step a cost function on the space of all possible partitions. Some other methods use more sophisticated models such as MRF [SGD84], but they tend to be quite inefficient in terms of computational time.

It is important to state that numerous works take advantage of MSTs as a mean to reduce the inherent algorithmic complexity. In [TVA93] vertexes connected by the smallest weight edge are melted by an iterative process. At the end, the MST formed at each step is further split by
removing edges bearing the highest weight while generating a hierarchy of partitions. In [YXE97] a MST is built up using the Kruskal's algorithm to find a partition minimizing a cost function afterwards. This is accomplished with a dynamic approach and diverse heuristics to further reduce the algorithm complexity.

The approach in [PFF98] is even more drastic, combining both region-growing and Kruskal's routine. Despite in [JSJ97] it is argued that in order to capture non local image properties a segmentation algorithm should start with large image regions and split them afterwards, rather than starting with small image regions and then merging them, [PFF98] proves that a region-merging algorithm can as well produce segmentations from non local image properties by a bottom-up scheme.

### 4.2.4 Edge Based Methods

Segmentation may also be obtained by detecting the edges among regions as it was extensively investigated for gray-level images [NPP93], from where it is well-known that edges can be found by using functions approximating the gradient or the Laplacian of an image, which are of course scalar functions. The problem encountered in color images is that of finding a counterpart of gradient functions for color images. This can be basically defined at least in two ways, namely, by embedding in a single measure the variations of all the three color channels, or by computing the gradient of each single channel and combining them accordingly to a given criteria afterwards.

The first approach requires some basic concepts of differential geometry such as the first fundamental form. Its eigenvectors provide the direction of maximal and minimal change while its eigen values provide the corresponding rates of change. The chromatic edge detectors in [MCA97] for vector-valued functions is based upon this metric. Numerous examples of the latter approach are given instead in [LLS99, LLS01], e.g., different combinations of gradients of hue, saturation, and intensity computed in HSI coordinates, or finding clusters in the RGB space and computing edges as the transitions from one cluster to another.

A truly original algorithm for boundary detection is proposed in [WYB97]. They use a kind of predictive coding model to identify the direction of change in color and texture at any point and at a given scale. This gives rise to an edge flow which, through propagation, converges to the image boundaries. There are several arguments in favor of hue as the most important color attribute for
segmentation. In particular, the work in [FPC94] demonstrates that, if the integrated white condition holds, hue is invariant to certain kinds of highlights, shadings, and shadows. Edge detection is then achieved by finding the zero crossings of the convolution over the hue image with a suitable Laplacian function, as in the gray-level case. Nevertheless, the poor behavior of hue near small values should be taken into account in that situation. Neural networks in the form of Kohonen's self-organizing maps can also be used for contour segmentation, as reviewed in [LLS99, LLS01].

The framework for object segmentation based on color snakes or active contours, originally proposed in [MKA87], can also fit within the context of edge-based techniques. The classical snakes approach consists in deforming an initial contour towards the boundary of the detected object. Deformation is obtained by minimizing a global energy such that its local minimum is attained at the boundary of the object. Formulation of active contours for vector-valued images and, therefore, for color images, due to Sapiro [GSV96, GSC97], is based on a Riemannian metric which captures information from all image components. Instead, color-invariant snakes that use color-invariant gradient information to drive the deformation process are proposed in [TGS98]. In this way, snakes return region boundaries pretty insensitive to disturbances due to shadowing, shadows, and highlights. Papers in [GSV96, GSC97] likewise show a close relation existing between active contours for color images and other algorithms based on frameworks such as partial differential equations, anisotropic diffusion, and variational approaches to image segmentation.

4.2.5 Neural Network Method
Finally, in [LLS99, LLS01] it is cited the class of image segmentation techniques adopting a classification based on neural networks. It is well-founded that neural networks are structures made up of a large number of elementary processors massively interconnected performing simple functions each. Despite their complexity, neural networks offer two important properties in pattern recognition tasks, namely, high degree of parallelism, which allows very fast computational times and makes them suitable for real-time applications, and good robustness to disturbances, which provides reliable estimates.

Another interesting feature is that, in the case of image segmentation, neural networks permit accounting for spatial information. On the other hand, in most kind of networks the final number of segments within an image must be known beforehand and run a preliminary learning phase to
CHAPTER 4. SEGMENTATION

train the network to recognize patterns. Usually segments are derived with some a priori knowledge about the problem or in a preprocessing stage.

A number of algorithms were already proposed in [NPP93] for segmenting gray-level images by means of neural networks. What is new in the reviews in [LLS99, LLS01] is that the discussion on neural-network based techniques is just offered in the field of segmentation of color images.

The authors in [PCD97] present two algorithms based on the idea of regarding segmentation as the problem of minimizing a suitable energy function for a Hopfield network. The first algorithm consists of three different networks, each dedicated to a color feature, combining the results afterwards. The second algorithm consists instead of a single network which classifies image pixels into the classes obtained by a preliminary histogram analysis in the color space. Other slightly different versions are also cited in [LLS99, LLS01], e.g., one using a pre-classification algorithm to spot out some regions of interest in biomedical images and another one applying an active-region segmentation algorithm.

It is important to state that this kind of techniques is optimal whenever the specific classification problem is well understood and the number of possible classes is beforehand known. This is quite the case both in medical applications and in the issue of human face localization by means of color segmentation.

An example for the latter is the work in [HAR98], where a retinally connected neural network examines small windows of an image and decides whether each window contains a face. The system arbitrates between multiple networks to improve the performance over a single network and a bootstrap algorithm is employed for training. False detections are added into the training set as training progresses in order to eliminate the need of a manual selection of negative training samples, which must be chosen in order to span the entire space of non face images.

In [HCF00] a neural network-based scheme for human face detection and eye localization in color images under an unconstrained scene is presented. A Self-growing Probabilistic Decision-based Neural Network (SPDNN) is used to learn the conditional distribution for each color class. The paper demonstrates a successful application of SPDNN to face detection and eye localization on a populated database as well as a good processing speed.
CHAPTER 4. SEGMENTATION

Regarding the field of medical images, a swell of algorithms has also been proposed dealing with the segmentation of color images. For instance, an algorithm for medical stained images is presented in [HON94], where three are the possible classes represented by three different colors. They suggest a three-layered neural network as the input layer and the three desired classes as the output layer.

In this regard, it is very common in neural networks the adoption of three layers since this structure is capable of implementing arbitrarily complex decision surfaces composed of intersecting hyper-planes in the pattern space. Classical also are the learning phases obtained with a back propagation routine as in [SNK97]. Similarly, the paper in [NFF94] uses two three-layered neural networks along with the learning through back propagation to separate cells from background in medical images.

In [MSR99] an unsupervised approach using Hopfield neural network is presented for the segmentation of color images of stained liver tissues. As in [PCD97], the segmentation problem is formulated then as the minimization of an energy function, with the addition of some conditions to reach a status close to the global minimum in a pre-specified time of convergence.

Recently, an efficient and accurate tool for segmenting color images has been proposed in [DGM02] grounded on a cluster-based approach to train very large feed-forward neural networks. This paper shows a great potential in applications where the accuracy is the major factor, specially in the area of medical imaging, where segmentations must provide the highest possible precision.

4.3. Physics Based Methods

All the algorithms examined so far are certainly prone to segmentation errors if the objects portrayed in the color images are affected by highlights, shadowing, and shadows. These phenomena cause the appearance of color of uniformly colored surfaces to change more or less drastically, whence those algorithms are very likely to return over segmented regions. The only way to overcome this drawback is to analyze how light interact with colored materials and to introduce models of this physical interaction in the segmentation algorithms. This motivates the name of physics based techniques given to them. The mathematical tools they use do not significantly differ from those adopted by the algorithms of the previous section; the major
CHAPTER 4. SEGMENTATION

difference with respect to those is the underlying physical model accounting for the reflections properties of colored matter.

Colored materials may be divided into three main categories: optically inhomogeneous dielectrics, optically homogeneous dielectrics, and metals. A milestone in the field of physics based segmentation was laid by Shafer in [SAS85] where he introduces the dichromatic reflection model for inhomogeneous dielectrics. This model is defined by

\[ \zeta(\lambda; g) = \zeta_s(\lambda; g) + \zeta_b(\lambda; g) = m_s(g)c_s(\lambda) + m_b(g)c_b(\lambda) \]

and states that the total radiance \( \zeta(\lambda; g) \) of the light reflected by an inhomogeneous dielectric is given by the sum of two independent parts: the radiance \( \zeta_s(\lambda; g) \) of the light reflected by the object's surface and the radiance \( \zeta_b(\lambda; g) \) of the light reflected from the underlying object's bulk. Symbol \( g \) denotes dependence on geometric parameters while \( \lambda \) is the wavelength. Moreover, the dichromatic reflection model states that each of the previous components can be split into a pure geometric coefficient \( m(g) \) independent of wavelength and into a relative spectral power distribution \( c(\lambda) \) that depends on wavelength but not on geometry. Shafer proves that in a color space such as the RGB the dichromatic reflection model simply reads

\[ C_\zeta = m_sC_s + m_bC_b, \]

where \( C_\zeta \) is the color (pixel value) measured, \( m_s \) and \( m_b \) are the magnitudes of reflection at the considered point, and \( C_s \) and \( C_b \) are the colors of interface and body reflection of the material. This model may effectively explain some particular shapes of clusters in the color space. Based upon this model, Klinker et al. [GJK90] set up an algorithm (using either a split or a region-growing strategy) which makes some optical hypotheses relating objects' colors, shading, and highlights and try to justify with them the cluster shapes. The main limitation of this technique is that it can be applied only to inhomogeneous dielectrics.

Figure 4.1(a). Natural Image

Figure 4.1(b). Natural Image (Segmented)
CHAPTER 4. SEGMENTATION

Figure 4.1(c). Cluster 1

Figure 4.1(d). Cluster 2

Figure 4.1(e). Cluster 3

Figure 4.1(f). Cluster 4

Figure 4.1(g). Cluster 5

Figure 4.1(h). Cluster 6
CHAPTER 4. SEGMENTATION

4.4 Discussion

From the above discussion we have come to know that segmentation is the process of partitioning an image into disjoint and homogeneous regions or by finding edges or boundaries between different color regions. The homogeneous regions or the edges correspond to actual objects or parts of objects within the image. The regions should be spatially well connected i.e. having spatial relationship and homogeneous in a certain feature space such as color or texture. The edges define a sharp boundary between objects or parts of objects. Adjacent regions of a segmentation should have significantly different values.

A natural image is constituted of many different color distribution that forms segments. This distribution of color overlaps or diffuses with each other and cannot be demarcated by a fine edge or boundary separating them. In this case, one of the property of segmentation, i.e. disjoint color regions cannot be achieved. Another important property of segmentation is connectedness of homogeneous (say same colored pixels) regions. In this case let a particular colored pixels be concentrated in two different places of the image quiet a distant apart such that they can not be put in the same segment due to a distance factor. These two anomalies are taken care of in the
methods discussed in Image-Domain Based Techniques. But none of those techniques operate in linear time.

Figure 4.1(a) depicts a natural image and Figure 4.1(b) depicts a segmented image of image Figure 4.1(a). Figure 4.1(c) to Figure 4.1(k) depicts the clusters that forms the objects in the image of Figure 4.1(a). It can be clearly seen that the color clusters that forms the objects does not have a sharp boundary between color regions demarcating an edge to form a segment. To get a smooth boundary between segments a smoothing algorithm has to be run which may detoriate the segment formation. Hence it will be less erroneous if we choose color clusters that forms the objects or parts of the objects. In our example image Figure 4.1(a), the combination of clusters Cluster 4, Cluster 6, Cluster 7 and Cluster 8 forms the hill object of the example image. This hill object does not have a sharp boundary demarcating it from the background.

We have developed a sliding window based technique which has been discussed in the next chapter to identify color clusters of any shape within the images and it operates in linear time.