Chapter 3

State-of-the-Art Survey on Color Segmentation Methods

3.1 Introduction

Not until recently segmentation techniques were mainly proposed for gray–level images on which rather comprehensive surveys can be found in [FM81, HS85, PP93]. The reason is that, although color information permits a more complete representation of images and more reliable segmentations, processing color images requires computational times considerably larger than those needed for gray–level images. Nevertheless, this is no longer a major problem due to the increasing speed and decreasing costs of computation. Besides, relatively inexpensive color cameras are easily available nowadays.

In accordance to these reason, there has been a remarkable growth in the number of algorithms that segment color images in the last decade [SK94, LM99, LM01, CJSW01]. Most of the times, these are kind of *dimensional extensions* of techniques originally devised for gray–level images. Thus, they exploit the well–established background laid down in that field. In other cases, they are *ad hoc* techniques tailored on the particular nature of color information and on the physics driving the interaction between light and colored materials.

In relation to our main concern in this work, namely, the segmentation of color images for robotics purposes, it must be said that image segmentation is a essential but critical component of any image analysis and/or pattern recognition system. Besides, it is one of the most difficult tasks in image processing, and determines the quality of the final results of the image analysis. In short, segmentation is defined as the process of partitioning an image into disjoint and homogeneous regions. This task is equivalently achieved by finding the boundaries between image regions.

The desirable characteristics that a good image segmentation should exhibit with reference to gray–level images were clearly stated in [HS85]: *"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 precise definition of segmentation accounting for the principal requirements listed above is given in [FM81, PP93] in the following way. Image segmentation is a process of dividing an image into different regions such that each region is homogeneous, but not the union of any two adjacent regions [PP93]. Nevertheless, according to [FM81], *"the image segmentation problem is basically one of psychophysical perception, and therefore not susceptible to a purely analytical solution".*

The rest of the Chapter is completely devoted to the review of the most important algorithms for color image segmentation. Owing to the extremely wide extension of this issue, it is nearly impossible to refer here to all the works found and read. Therefore, on behalf of a clearer exposition, this Chapter is divided into three main periods, namely, *early*, *middle*, and *recent* stages. Besides the long–standing readings done to keep pace with the color segmentation issue, this review has also been grounded on the small group of former surveys [PP93, SK94, LM99, LM01, CJSW01] found on that subject, which has been of great help. At the end, some conclusions drawn about the issue are summarized.

3.2 Early Stages

As said before, in these early years most of the segmentation approaches were mainly devoted to gray–level images. Monochrome segmentation is based on discontinuity and/or homogeneity of gray–level values in a region. The approaches based on discontinuity tend to partition an image by detecting isolated points, lines and edges according to abrupt changes in gray levels. The approaches based on homogeneity include thresholding, clustering, region–growing, and region splitting and merging.

The work in [FM81] divides the existing segmentation techniques into three categories, i.e., clustering or characteristic feature thresholding, edge detection, and region extraction. Because this is a review eminently versed on medical images, very scarce information is given on the segmentation of color images. Threshold selection schemes based on gray–level histogram and local properties, along with structural (textural) and syntactic techniques are described in [FM81]. Clustering techniques are regarded as the multidimensional extension of the concept of thresholding. Some further clustering schemes utilizing different kinds of features (multispectral information, mean/variation of gray–level, texture, color) are also discussed in that paper.

Additionally, various techniques for edge detection are presented for both parallel and sequential techniques. In parallel techniques, the decision on whether or not a set of points belong to an edge depends on the gray level of the set and some of its neighbors. These techniques include spatial frequency filtering, gradient operators, adaptive local operators, functional approximations, heuristic search and dynamic programming, relaxation, and line and curve fitting. Sequential techniques take decisions based on previously examined points, giving as a result a brief description of the major component in the sequential edge detection. Region merging and splitting approaches are also considered there.

Segmentation Techniques	**Advantages**	**Disadvantages**
Histogram Thresholding	• Does not need prior information of the image • For a wide class of images satisfying the requirement, this method works very well and has low computation complexity	• Does not work well in images without obvious peaks or valleys • Does not consider spatial details, so it cannot guarantee that segmented regions are contiguous
Feature Space Clustering	• Straightforward for classification and easy for implementation	• How to determine the number of clusters (known as cluster validity) • Features are often image dependent and how to select features to obtain satisfactory segmentations remains unclear • No spatial information is utilized
Region–Based Approaches	• Work better when the region homogeneity criterion is easy to define • Are also more immune to noise than edge detection approaches	• Are sequential and quite expensive both in computational time and memory • Region–growing has an inherent dependence on the selection of seeds and the order in which pixels and regions are examined • Resulting segments by region splitting appear too square due to the splitting scheme

Table 3.1: Monochrome image segmentation techniques.

Segmentation Techniques	Advantages	Disadvantages
Edge Detection Approaches	• Edge detection is the way in which human perceives objects and works well for images having good contrast between regions	• Do not work well with images in which the edges are ill–defined or there are too many edges • It is not a trivial job to produce a closed curve or boundary • Less immune to noise than other techniques, e.g., thresholding and clustering
Fuzzy Approaches	• Fuzzy membership function can be used to represent the degree of some properties or linguistic phrases, and fuzzy `if-then` rules can be used to perform approximate inferences	• The determination of fuzzy membership is not a trivial job • Computations involved in fuzzy approaches can be intensive
Neural Network Approaches	• Do not need complex programs. Can fully utilize the parallel nature of neural networks	• Training time is long • Initialization may affect results • Overtraining should be avoided

Table 3.2: Monochrome image segmentation techniques (Cont.).

Subsequently, in [HS85] several segmentation techniques are classified into the categories of spatial clustering driven by a space of features, different kinds of region–growing algorithms depending on the way regions are joined – single linkage, hybrid linkage, and centroid linkage –, spatial clustering, and split&merge. The survey presents some spatial clustering approaches which combine clustering in feature space with region–growing or spatial linkage techniques. It gives a good summary of different kinds of linkage schemes for region–growing methods. The problems of high correlation and spatial redundancy of multi–band image histograms and the difficulty of clustering using multidimensional histograms are also discussed. However, this survey fails in extensively tackling the color image segmentation issue.

The review of segmentation techniques in the basic work of [PP93] is mainly centered again in images that can be reduced to its gray–level values, but rather this time a small survey into specific color segmentation algorithms has also been included. In addition, the work in [PP93] criticizes former surveys – [FM81] and

[HS85] – for two main reasons. First, neither fuzzy–based segmentation techniques nor neural networks–based approaches are taken into account. Moreover, the issue of segmentation evaluation is neither considered nor there is any specific method for color images.

All those aforesaid strategies are also reviewed in [PP93] at the same time as gray–level thresholding, edge detection, and MRF–based[1] and surface–based approaches. It also considers range images and magnetic resonance images (MRI). After the discussion of these segmentation approaches, the authors make a comparison of six histogram–based methods and two iterative pixel classification methods, based on relaxation and *Maximum A Posteriori* (MAP) estimation, respectively. Finally, attempts for a quantitative evaluation of segmentation results are studied too. Tables 3.1 and 3.2 briefly summarize all those techniques. Their importance here dwells in the fact that most of the algorithms for color segmentation are fully based on some preceding gray–level methods.

In regard to the precise issue of color image segmentation, only the review in [PP93] encompasses seven references. This survey says that a number of linear – as those in [OKS80] – and nonlinear transformations can be adopted as color features. In addition, three algorithms are described, that in [OKS80], which is based on a former recursive region splitting algorithm, and the one in [LL90], consisting in the application of a threshold followed by a fuzzy k–mean method. Finally, there exists the algorithm in [HJC85], which is similar to the previous one but uses a space–scale filter to find the initial number of clusters.

3.3 Middle Stages

Unlike the paper [PP93], the survey in [SK94] is an exhaustive and extensive review of algorithms to segment color images which represents an answer in front of the high number of reviews on the segmentation of gray–level images and the seemingly existing desert of color counterparts. This survey spans 81 references in 80 pages, where 51 different methods are surgically examined. Also in this survey a nourished revision on color spaces and a list of interesting conclusions about the issue are furnished.

In [SK94] the segmentation process is defined as the extraction of a group of connected segments from the spatial domain satisfying a uniformity criterion – homogeneity defined in [PP93] – based on the color characteristics of the image within a chosen color space. This process could be further augmented by joining information about the objects in the scene, such as shape or surficial properties. The most important feature in any segmentation process is what is understood for a segment. Accordingly to [SK94], there are at least four typologies

- A connected component from a set of pixels specified by a membership function defined in a color space.
- A maximal connected set of pixels satisfying a uniformity condition.
- A connected set of pixels surrounded by border pixels forming a color contour.
- A surface or an object made of a uniform matter.

[1] Markov Random Field.

The first two definitions of *segment* share kind of uniformity predicate, the former referred to pixels and the latter to image areas. Regarding the third definition, a nonuniformity predicate is employed. In the fourth case, an alternative class of regions is presented which aim is to tie a closer relation between color and what really exists in a scene, rather than being a mere extension of former definitions applied to gray–level images.

Therefore, the categories which segmentation algorithms are divided into in [SK94] are grounded on the previous definitions and correspond to *pixel–based*, *area–based*, *contour–based*, and *physics–based* methods, respectively. Second and third category had already been extensively dealt in literature [PP93], whereas the first one gathers the set of techniques based on histogram thresholding and clustering. The last definition employs reflection models which are based on the properties of materials in the scene. This is a pretty new approach hardly tackled in former surveys. Next, these categories will be briefly reviewed.

3.3.1 Pixel–Based Methods

Algorithms using color information are considered in three main classes

- *Histogram–based techniques:* Clusters are identified via finding peaks of frequency in the histogram. Thereafter, image pixels are classified as belonging to one of those classes thus formed. Regarding to similar approaches in gray–level images, color histograms have more dimensions than one, and the search for peaks can be done either independently in each color channel or in the whole 3D histogram.

- *Clustering techniques:* Pixels are grouped by means of their color values forming clusters whose prototypes are posteriorly employed in the classification of image pixels.

- *Fuzzy clustering techniques:* Several fuzzy membership functions are evaluated in [SK94]. The grouping of pixels in what is known as *crisp* clusters representing image segments are obtained by a *defuzzification* process and a subsequent subdivision into maximal connected components. A popular choice is the fuzzy k–means algorithm [Bez81].

3.3.2 Area–Based Methods

Survey in [SK94] divides algorithms based on region uniformity into two different classes

- *Region–growing techniques:* This strategy needs an initial set of seeds to work, as well as a general criterion to join neighboring regions. In order to distinguish these algorithms from the next ones, it is important to state that seeds do not result from any previous division process of nonuniform regions, rather from a selection.

  Nevertheless, these methods are known to be sensitive to the seed choice process together with the way segment statistics are computed, which is done to guess whether two adjacent regions might join or not. It is of special interest the set of graph–theoretical approaches, as will be seen later.

- *Split&Merge techniques:* These methods start from nonuniform regions which keep dividing until a uniformity criterion is satisfied. Posteriorly, regions thus obtained are recombined to achieve uniform regions as big as possible. The splitting phase is analogous to the aforementioned seed selection step and it is often carried out by analysing the image histogram. Whereas, the merging phase can be fulfilled by a region–growing method.

3.3.3 Contour–Based Methods

A said in [SK94], in essence, there are relatively few approaches using the detected contours straightforward to segment images, although those contour points are important in tasks such as stereo correspondence in color images. Here, contour techniques are separated in two categories

- *Local techniques:* Only the information about neighboring points is needed to know whether a pixel is located in a border or not. Usually, local contour detectors are faster than those in the next category, but it is not clear yet how to translate operators based on gradient and the Laplacian of gray–level images into their color analogy, despite several attempts have already been done in this direction [Cha92, Cum91].

- *Global techniques:* Most of these algorithms take into account global optimization processes mainly based on *Markov Random Fields* (MRF) [GG84, Li95]. That way, many previous optimization steps implying changes in wide areas are required in order to know if a given point belongs to a contour. These approaches are often slow to converge, although results are really good, specially in the noisiest areas in the image.

3.3.4 Physics–Based Methods

The goal of these methods is to segment images according to the real contour of objects avoiding being misled by shades and highlights in the image. This is a pretty difficult goal since measures coming from a single surface may vary in a great extent due to interreflections, shadows, shades, sensor noise, nonuniform illumination or texture surface [Hea92a].

Hence, if it worked, segmentation based on physical models would permit segmenting via the study of the process of light reflection and image formation. Algorithmically, the basics of these approaches are often very similar to those of other segmentation methods, and only differ from them in the fact that these algorithms explicitly use the reflectance models of surfaces to segment color images.

Therefore, it seems that these methods can only be applied in case the properties of surfaces are known to a great extent. Some schemes even want to be utilized before the segmentation itself, trying to distinguish among material changes from highlights and shades. Next, a brief review of these methods.

- *Inhomogeneous dielectrics materials:* There is a great number of reflection models, most of them developed in the field of computer graphics [TS67, Pho75, CT81, Wol94]. Among these methods, the *dichromatic reflection* model in [Sha85] is a usual choice for those algorithms employing a physical model to segment color images.

This model postulates that the light reflected from an infinitesimal piece of surface of an inhomogeneous dielectric material[2] is the result of the addition of two components, namely, the *interphase* (specular) reflection and the *body* (diffuse) reflection. Each of these parts is further divided into two elements, one accounting for the geometry and another purely spectral.

This way, the final color of a surface is made up of a linear combination of two colors, one from the surface and another from the body. To obtain the true color of the object, the specular component should be removed. Due to the particular shape – a plane – in the RGB space that colors from a material generate, some algorithms [Kli88, KSK88, KSK90] try to estimate the number of surfaces in an image as well as which pixels belong to each surface.

On the other hand, the works in [MS94, MS96, MS97] present a new approach to segmentation using explicit hypotheses about the physics creating the images. After proposing an initial segmentation that identifies image regions exhibiting constant color, a set of hypotheses modeling illumination, reflectance, and shape are proposed for each region. Hypotheses for adjacent patches are compared for similarity and merged when appropriate, resulting in more global hypotheses that group elementary regions.

Yet, the work in [Tom91] goes a step beyond and is capable of approximately estimating the spectral power distribution of the illuminant in order to remove it *a posteriori* from the color of objects in tasks of identification using images taken in an indoor environment.

- *General approaches:* Besides the above kind of methods, there are other approaches endeavouring the problem of finding general reflection models. In a special position there exist the works in [Hea89b, Hea89a, Hea90, Hea92b] whose main idea is to adopt a splitting region algorithm in a normalized color space. The important point there is color normalization, which takes into account the physical properties of both metallic surfaces and inhomogeneous dielectric materials.

3.4 Recent Stages

In [LM99, LM01], 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. On the other hand, possible applications of color segmentation are stretched to include *multimedia applications* such as image and video retrieval from digital databases, information transmission via the Internet, and the latest generations of mobile telephones, as well as image compression for television transmission.

These works classify segmentation techniques into three main categories

- Feature–based techniques.

[2] Plastics, paper, ceramic, wood, and fabric.

- Clustering.
- Adaptive k–mean clustering.
- Histogram thresholding.

• Image–based techniques.

- Split&Merge.
- Region–growing.
- Graph–theoretical techniques.
- Edge–based techniques.
- Neural networks.

• Physics–based techniques.

In the first two categories, subdivisions within are fully inspired in former reviews such as [PP93], basically focused on the segmentation of gray–level images, as can be observed from Sections 3.2 and 3.3. This kind of classification is not always useful since many methods share ideas from diverse categories. The last category has no counterpart in gray–level images and was already introduced by [SK94], as explained earlier.

3.4.1 Feature–Based Methods

The first group of segmentation methods described in [LM99, LM01] is the one based on the space of features, which can be further split into clustering methods, adaptable k–means clustering, and histogram thresholding. The idea behind those methods is that color is a constant feature of objects' surfaces so that in a certain color space it forms a distinguishable cluster or peak in a color histogram. The spreading within a cluster is mainly determined by color variations due to shading, highlights, and device noise. A common feature for all those approaches is that of completely neglecting the spatial information among of pixels sharing the same color.

Clustering Methods

Clustering can be broadly defined as a nonsupervised 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 [PP93], 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 [PYL98], widely adopted in vector quantization and data compression. A *fuzzy* version of this is commonly used in a number of works referred to in [LM99, LM01], as well as the closely related approach of *probabilistic* clustering. A comparison between *crisp* and *fuzzy* versions of that algorithm can be found in [RTT95]. ISODATA[3] is another algorithm often used for color space clustering [TA99].

Another interesting and fruitful approach is the *mean–shift* algorithm reported in [CM97, CM99, CM02]. 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 [CRM03].

Competitive learning based on the least–square criterion is employed in [UA94], whereas the theory of connected components is adopted in [WSC97]. An original technique proposed in [YL98] 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 [Uch94] 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 belive 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 [JD88, JDM00, MP00]. 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 [JD88], 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, MK97, MP00] 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 [FJ02] 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 [LFJ04] 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

[3] Iterative Self–Organizing Data Analysis Techniques.

in [Yam98] 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. No specific segmentation approaches for color images explicitly employing the EM algorithm in [FJ02] has been found in the literature yet. Nevertheless, we have extensively used such approach to obtain color segmentations both in Chapter 5 and in 7, attaining by the way pretty good results.

Adaptive Clustering Methods

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. In this sense, these algorithms lie in between the techniques based on feature spaces discussed here and the techniques grounded on the image domain that will be considered next. 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 [Pap92] 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 [CST94]. 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 [GG84, Li95] 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.

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 [STB96], where color image segmentation and edge linking are combined, applying a split&merge strategy to enforce edge consistency. Besides, in [LGL97, LGL98] the algorithm described in [CST94] 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.

Histogram Thresholding Methods

Histogram thresholding is among the most popular techniques for segmenting gray–level images and several strategies have been proposed [FM81, HS85, PP93]. 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 [CdH98]. In [SPK98] 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 oversegmentation. In [TLT95] 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 [SK97] a fast segmentation algorithm is suggested which resorts to a preclustered chromaticity plane after quantization of the HSV space represented into orthogonal Cartesian coordinates.

The work in [SP96] 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 [STEK95] as a Gaussian PDF allowing this way an adaptive setting of object–class thresholds. In [LLY$^+$94] 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 [LM98] by working with the low–low band of the wavelet transform of the image to be segmented.

3.4.2 Image–Based Methods

Surveys in [LM99, LM01] criticize the methods that are just grounded on feature spaces since no spatial relation is taken into account. Thus, the class of regions these algorithms usually return is expected to be homogeneous with respect to the characteristics represented by these spaces, but rather there is no guarantee of spatial compactness, which would be a second desirable property of segmentations, besides that of homogeneity.

In fact, cluster analysis and histogram thresholding account in no way for the spatial locations of pixels. Their description is global and does not exploit the important fact that points of a same object are usually close due to what is known as *surface coherence*. On the other hand, if pixels were exclusively clustered on the basis of their spatial relation, segmentations would be too local and the final result would be likely made of regions spatially well connected but with no homogeneity guaranteed.

As seen, in the literature of gray–level segmentations a heap of techniques has been suggested trying to satisfy both feature–space homogeneity and spatial compactness at the same time [HS85, FM81, PP93]. Depending on the strategy preferred for spatial grouping, these algorithms are classically divided into split&merge and region–growing techniques. This distinction has also been inherited in [SK94] by the corresponding algorithms that segment color images. It is similarly done in [LM99, LM01], where a family of algorithms exploiting neural–network classifiers and those partitioning images by finding edges between homogeneously colored regions have been included inside the class of image–based techniques. Moreover, for its relevance and interest for the present work, a special section is fully devoted to the set of graph–theoretical algorithms.

Split&Merge Methods

The description of these methods in [LM99, LM01] is similar to others found in previous reviews [PP93, SK94]. These methods start with an initially inhomogeneous partition of the image and they keep splitting until homogeneous partitions are obtained. A common data structure used to implement this procedure is the *quadtree* representation. After the splitting phase, there usually exist many small and fragmented regions which have to be somehow connected in a merging phase. The *Region Adjacency Graph* (RAG) is the data structure commonly adopted in this phase. In many algorithms, smoothness and continuity of color regions are enforced by the adoption of a *Markov Random Field* (MRF) [GG84, Li95, Wan98], which is equivalent to a *Gibbs* distribution.

Color texture in RGB is modeled by a *Gaussian Markov Random Field* (GMRF) in [PH95] embedding the spatial interaction within each of the three color channels as well as the interaction between them. In [LY94] a MRF is defined instead on the quadtree structure of the color image and the equivalence to the Gibbs distribution is applied. A *relaxation* process controls both splitting and merging of blocks in order to minimize the energy of the Gibbs distribution. This is shown to converge to a MAP estimate of the segmentation.

Numerous variations of the split&merge strategies have been investigated. A k–means algorithm in [Cel97] for both classifying pixels in the splitting phase and grouping pattern classes in the merging phase. In [JP98] the splitting is operated by the watershed transform of the gradient image of the luminance component, simplified by a morphological opening. The merging step is carried out using a Kohonen's Self–Organizing Map (SOM).

Alternatively, in [SPK97] the watershed transform is applied to the gradient of L*u*v* images and patches of the watershed mosaic are merged according to their color contrast until a termination criterion is met. A similar splitting approach is adopted in [Saa94], where the merging phase is performed by iteratively processing the RAG upon the resulting oversegmented regions. In [BRM96] the split&merge strategy is implemented by a fuzzy expert system.

Some authors believe that split&merge algorithms based on quadtrees are not capable of adjusting their tessellation to the underlying structure in the image because of its rigid rectilinear nature [GS97]. Therefore, they suggest replacing it with an incremental Delaunay's triangulation. A further alternative possibility is to use the Voronoi's diagrams, as proposed in [SS94]. Instead, in [RP97, RP98] images are represented into a hexagonal connectivity using a hierarchical tree structure known as the *Color Structure Code* (CSC) algorithm, specially adapted for performing real–time segmentation of natural color images.

Broadly speaking, we can fit within the class of split&merge techniques also some algorithms based upon differential equations and pyramidal data structures. At first glance, they appear not to belong to this category. However, a more careful look into them raises the same underlying ideas of the split&merge algorithms.

Region–Growing Methods

There is nothing new in obtaining a homogeneous region from an image through a growth process which, starting from a preselected 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. A common post–processing provision consists of a *merging phase* that eliminates small regions or neighboring regions with similar attributes, generating broader regions accordingly.

Region–growing can be considered as a sequential clustering or classification process. Thus, the results may depend on the order according to which image points are processed. The main advantage offered by this kind of techniques is that regions obtained are certainly *spatially connected* and rather *compact*. Nevertheless, as for the aforementioned clustering techniques, the problems of choosing suitable seed points and an adequate homogeneity criterion also come up for the case of region–growing techniques.

For color images, most strategies mimic those of gray–level and there are few real novelties in regard to the approaches reviewed in [SK94]. Apart from some differences on the way seeds are found and refined (usually a quantization process or some extrema finding method), a typical growing scheme is that of *watershed*. To reduce the number of final regions, a merging stage is commonly undergone.

The work in [TB97] suggests several different homogeneity criteria driving the growth process in a first phase and obtains a certain number of connected regions, whereas components having similar color distributions are merged in a second phase. [Kan98] develops an algorithm which resorts to both color and intensity information. Markers (seeds) are extracted from intensity by way of morphological open/close operations and also from color through quantization of the HSV space. Joint markers arise from comprising both kinds of markers and they are used afterwards to start the region–growing process based on a watershed algorithm.

In [CGA97] initial seeds are generated by retaining significant local minima in the color image gradient and a posterior procedure is devised for obtaining just one marker per image region. The region–growing is then performed with a watershed–like algorithm proposed by the authors that works in the original color image instead of in a gradient image.

The article in [DMS99] determines a limited number of color classes within an image through color quantization and proposes a criterion for a *good* segmentation that is based on them. When this criterion is applied within local windows and at multiple scales generates high values over possible region boundaries and low values over region centers. A region–growing method is then adopted starting from the valleys, taken as seeds. The resulting oversegmentation is finally removed by a merging phase.

The interesting work in [RP97, RP98] suggests a robust and fast method using a special hexagonal topology in a hierarchical region–growing algorithm which happens to be independent from the starting point and the order of processing. Besides, the paper [MB97] has adopted a fuzzy outlook of the problem and advances two algorithms that work in the RGB space and implement a region–growing strategy for both fine and coarse segmentations of color images.

Finally, we cite the work in [CL97] which compares the results of four different algorithms obtained by several combinations of region–growing and watershed transform in a presegmentation step that is followed by such kind of algorithms.

Graph–Theoretical Methods

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 superfluous 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 [Zah71] is based on breaking large edges in a *Minimum Spanning Tree* (MST) of the graph. A more recent method [WL93] 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 *nonlocal* 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 [SM97, SBLM98], where a *normalized* version of the minimum cut is computed. For a wider review on this sort of approaches, we refer to the works in [Els97, Fja98].

In [Urq97] 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 nonlocal properties. This issue is specifically tackled in [FH98a]. Another graph–theoretical work in [Wan98] 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 [GG84], 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 [VC93] 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 [XU97] 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 [FH98a] is even more drastic, combining both region–growing and Kruskal's routine. Despite in [SM97] it is argued that in order to capture nonlocal 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, [FH98a] proves that a region–merging algorithm can as well produce segmentations from nonlocal image properties by a bottom–up scheme.

Edge–Based Methods

Segmentation may also be obtained by detecting the edges among regions as it was extensively investigated for gray–level images [PP93], 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 eigenvalues provide the corresponding rates of change. The chromatic edge detectors in [Cha97] for vector–valued functions is based upon this metric. Numerous examples of the latter approach are given instead in [LM99, LM01], 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 [MM97b]. 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 [PK94] demonstrates that, if the *integrated white condition*[4] 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 behaviour 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 [LM99, LM01].

The framework for object segmentation based on *color snakes* or *active contours*, originally proposed in [KWT87], 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 [Sap96, Sap97], is based on a Riemannian metric which captures information from all image components.

[4]For equi–energy white lights, areas under the curves of the three color channels are equal.

Instead, color–invariant snakes that use color–invariant gradient information to drive the deformation process are proposed in [GGS98]. In this way, snakes return region boundaries pretty insensitive to disturbances due to shadowing, shadows, and highlights. Papers in [Sap96, Sap97] 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.

Neural Network Methods

Finally, in [LM99, LM01] 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 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 [PP93] for segmenting gray–level images by means of neural networks. What is new in the reviews in [LM99, LM01] is that the discussion on neural–network based techniques is just offered in the field of segmentation of color images.

The authors in [CMS97] 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 [LM99, LM01], e.g., one using a preclassification 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 [RBK98], 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 *nonface* images.

In [FLLP00] 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.

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 [OKHO94], 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 hyperplanes in the pattern space. Classical also are the learning phases obtained with a *backpropagation* routine as in [KSPA97]. Similarly, the paper in [Fun94] uses two three–layered neural networks along with the learning through backpropagation to separate cells from background in medical images.

In [SSNM99] an unsupervised approach using Hopfield neural network is presented for the segmentation of color images of stained liver tissues. As in [CMS97], 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 prespecified time of convergence.

Recently, an efficient and accurate tool for segmenting color images has been proposed in [GYB02] 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.

3.4.3 Physics–Based Methods

Concerning the segmentation methods that are founded on physical models of light interaction with colored surfaces, conclusions in [LM99, LM01] are analogous to those drawn in [SK94]. All the algorithms examined so far are certainly prone to segmentation errors if objects portrayed in the color images are affected by highlights, shadowing, and shades, as it is the usual case. These phenomena cause the appearance of uniformly colored surfaces to change more or less drastically, whence those algorithms are very likely to return oversegmented regions. The only way to overcome this drawback is to introduce models of the physical interaction in the segmentation algorithms accounting for the reflections properties of colored materials.

Reviews in [LM99, LM01] divide colored materials into three main categories similarly as previously stated in Section 3.3.4, namely, optically inhomogeneous dielectrics, optically homogeneous dielectrics, and metals.

A milestone in the field of physics–based segmentation is the work in [Sha85], where the *dichromatic reflection* model for inhomogeneous dielectrics is proposed. This model has already been introduced in Section 3.3.4. The simplicity and effectiveness of that representation have made the dichromatic reflection model very popular and many physics–based techniques for segmentation resort to it. The main limitation of this technique is that it can be applied only to inhomogeneous dielectrics.

A very major contribution related to the previous model is presented in [BLL96], where color reflection based on the dichromatic model and on a particular color space, called S space, is proposed. In this space, brightness, hue, and saturation may be defined to analyze color variations. It is proven that it is possible to separate *specular* and *diffuse* interface reflections, and some inter–reflections from body reflections, since they produce clusters with very peculiar shapes in the S space. The algorithm in [BLL96] allows the segmentation of uniformly colored dielectric surfaces under singly colored lights.

Additionally, a *unichromatic* reflection model for metals is put forth in a number of papers [Hea89b, Hea89a, Hea92a, Hea92b] by supporting it with extensive experimental results. Such a model states that metals give rise to a reflectance function which stems only from their surfaces and which, analogously to the dichromatic reflection model, can be separated into a geometric factor and a purely spectral component. This way, geometric effects in a scene can be factored out by kind of a color normalization. Besides, these works come up with some segmentation algorithms based on such a normalization coping with both inhomogeneous dielectrics and metals at the same time.

The methods discussed above are able to work with one or two classes of materials − inhomogeneous dielectrics or metals − in the presence of a single illumination. A more general and more involved algorithm, which also accounts for multiple illuminations, is presented in [MS97], where a broad framework is introduced for the segmentation of complex scenes, which formulates multiple physical hypotheses about image formation. These hypotheses define classes for shape, illumination, and material properties of simple image regions obtained through an initial rough segmentation. A set of possible segmentations is generated by analyzing, merging, and filtering the hypotheses and, finally, pruning them, which yields a restricted number of plausible segmentations of the scene at the end.

3.5 Conclusions

One important fact has to be drawn from the present survey on color image segmentation. There is no universal theory on color image segmentation yet, and alike is the situation for general image segmentation. All of the existing approaches are, by nature, *ad hoc* to some extent. Most techniques are tailored on particular applications and may work only under certain hypotheses. So, to the problem of what segmentation method should be utilized, there is no clear answer. It depends on the application and our experience. Some of the advantages and disadvantages of such class of techniques are summarized in Table 3.3 and Table 3.4.

An image segmentation problem is basically one of psychophysical perception, and it is essential to supplement mathematical solutions by an *a priori* knowledge about the image. It has been reiteratively stated through this Chapter that most algorithms are extensions of former gray–level techniques, such as histogram thresholding, clustering, graph–partitioning, region–growing, edge detection, and fuzzy approaches. However, color allows more reliable image segmentations than gray–level, since more information is obviously provided.

Segmentation Techniques	Advantages	Disadvantages
	Feature–Based Techniques	
Clustering	• Nonsupervised classification • Existence of parametric and heuristic approaches	• Neglects spatial information • Problems in determining the initial number of classes • Difficulties to adjust classes to image regions
Adaptive Clustering	• Properties of local adaptivity to regions and spatial continuity • Spatial constraints are imposed	• Maximizing a posteriori probability can be misled by local maxima • Slow convergence
Histogram Thresholding	• No prior image information is needed • Easy and fast algorithms	• Neglects spatial information • Thresholding in multidimensional spaces is a complex task • Noise give rise to spurious peaks and extrema ambiguities
	Physics–Based Techniques	
	• Alleged robustness to highlights, shadowing, and shades • Segment surfaces by their material composition	• Restriction to one or two types of material, such as dielectrics or metals • Difficulty to identify highlights and shades in real images • Some algorithms need shape information, not always available • Computationally expensive algorithms

Table 3.3: Color image segmentation techniques.

Segmentation Techniques	**Advantages**	**Disadvantages**
	Image–Based Techniques	
Split&Merge	• Spatial information is taken into account • Good results for images with homogeneous regions	• Defining color homogeneity may present difficulties • Quadtree may generate artifacts
Region–Growing	• Spatially connected and compact regions are generated • Some approaches attain pretty fast algorithms	• Expensive both in computational time and memory • Difficulties in gathering a set of seeds and an adequate homogeneity criterion • Sequential nature of those algorithms
Graph–Theoretical Techniques	• Very good translation of spatial and feature relations into graph–based representations • Some greedy approaches provide very fast algorithms	• Some approaches are extremely time consuming • Local features may be imposed over global ones
Edge–Based Techniques	• Color reinforces edge detection • Good performance of active contours in object detection	• Difficulty in defining a color counterpart for the gradient function • Noise and poorly contrasted images badly affect edge detection
Neural Networks	• High degree of parallelism and very fast computational times • Good robustness to disturbances which allows reliable estimates • Efficient tools for specific applications, such as face detection or medical images	• Color may increase network complexity • Learning phase needs to know the number of classes beforehand

Table 3.4: Color image segmentation techniques (Cont.).

As a rule, the most clearly specified, algorithmically efficient, and robust methods are designed for particular small applications assuming well specified knowledge about the scene. Conversely, general purpose algorithms are neither robust nor usually algorithmically efficient. It seems that separating processes for region segmentation and for object recognition is the reason of failure of general purpose segmentation algorithms. Unfortunately, the authors in most of the above works do not estimate the algorithmic complexity of their methods and often ignore comparing their novel ideas with existing ones.

One and capital problem of color segmentation is how to employ color information as a whole for each pixel. When color is projected onto three components, color information is so scattered that color images become simply multispectral images and the kind of information that humans can perceive is partially lost. As a consequence, a particular segmentation method can be directly applied to each component of the color space independently, or on the contrary to work with color as a whole representation. For the first option, problems arise when trying to combine the partial results in some way to obtain the final segmentation, whereas for the latter, the difficulty is that of finding a useful color space and metric to compare color differences.

In regard to the choice for an appropriate color space, we reiterate that it is entirely an image/application–dependent question. There is not any color space which is better than others and more suitable to all kinds of images yet, since each color representation has its advantages and disadvantages. Despite some authors claim that HSI–like color spaces can solve to some extent some of the problems related with color – except for the fact that hue is unstable at low saturations –, our big bets are on for the *perceptual uniform color* spaces, such as CIE L*a*b* [WS82, Fai97], just in case straight RGB turns out not to be enough. The idea behind these spaces is to adopt a *nonlinear* transformation in order to use an Euclidean distance to compare colors. A further main problem that comes up when combining region segmentation and color object recognition is the need for a *color constancy* stage to avoid color from changing when scene illumination varies, a main concern in our work.

On the other hand, in most color segmentation approaches, the definition of a region is based on the color similarity between neighboring pixels or on the color homogeneity of regions. These assumptions often make it difficult for many algorithms to separate regions in objects with highlights, shadows, shading, or texture, causing inhomogeneities in surfaces. Some physics–based models have been proposed to find the boundaries of objects on the basis of the type of materials. Unfortunately, nearly all these approaches can only be applied to a restricted set of materials, which limits their extensive application, besides being computationally expensive. Thus, they are far from being a general solution to image segmentation, nor suitable for real–time segmentation.

Finally, let us indicate that some authors [LY94, HD95, BCS98, Zha96, Zha97, Zha01, CW04] have proposed some heuristic measures for the quantitative evaluation of segmentation. Even in [CGVLS00] one of these measures is used to drive the segmentation process. However, the goodness of a segmentation depends on so many factors, such as homogeneity, spatial compactness, continuity, or correspondence with the psycho–visual perception, that a single measure is uncertain to capture all of them in a meaningful way. Such goodness should be evaluated by the effectivity of a given algorithm in the particular application one is interested in.