

Jaime dos Santos Cardoso

METADATA ASSISTED IMAGE SEGMENTATION



Departamento de Engenharia Electrotécnica e de Computadores
Faculdade de Engenharia da Universidade do Porto
Janeiro de 2006

Jaime dos Santos Cardoso

METADATA ASSISTED IMAGE SEGMENTATION



*Tese submetida à
Faculdade de Engenharia da Universidade do Porto
para obtenção do grau de
Doutor em Engenharia Electrotécnica e de Computadores*

Dissertação realizada sob a supervisão do
Professor Doutor Luís António Pereira de Meneses Corte-Real
Departamento de Engenharia Electrotécnica e de Computadores
Faculdade de Engenharia da Universidade do Porto
Janeiro de 2006

always to *Tina* and *Bino*, my parents
and to *Maria do Carmo*

Resumo

Até aos dias de hoje o processo de captura de imagens digitais tem-se baseado sempre no mesmo princípio, em que cada elemento da imagem é uma amostra discreta, no espaço e no tempo, da realidade contínua a capturar. Contudo, as imagens digitais são uma amostra muito limitada da realidade que representam, dando apenas uma impressão 2D da disposição 3D dos objectos. Este processo limitado de captura é fonte de problemas em inúmeras aplicações, desde a pós-produção cinematográfica à realidade virtual, passando pelos formatos vídeo baseados na codificação de objectos e pela inspecção industrial. Embora diversas técnicas tenham sido desenvolvidas para ultrapassar estes problemas, a verdade é que todas elas se baseiam na estimação de dados que simplesmente não foram capturados e, por isso, estão desde logo limitadas na qualidade que podem atingir. A captura de informação adicional surge como a solução mais natural para ultrapassar estes problemas.

Sendo a segmentação de uma imagem uma operação recorrente nas mais diversas áreas, e de cujo desempenho depende fortemente o desempenho das operações subsequentes, o seu estudo torna-se obrigatório. A informação de profundidade síncrona com a informação da cor deverá permitir atingir desempenhos significativamente superiores aos possíveis com as técnicas actuais, viabilizando toda uma série de novas aplicações.

Esta dissertação concentra-se no estudo de técnicas de segmentação de imagem assistidas por metadados. Em particular, investiga-se a utilização de informação de profundidade e movimento para melhorar a qualidade das segmentações de imagens a cores. Partindo de técnicas primitivas de fusão da cor e da profundidade, o trabalho evolui para técnicas mais ricas, com a modelação conjunta da cor e profundidade; este método é extendido para tratar de forma conveniente o ruído tipicamente presente na informação de profundidade em condições reais. Posteriormente é proposta uma abordagem alternativa, na qual a profundidade é utilizada para fornecer uma estimativa grosseira dos objectos na imagem, e a cor é utilizada de seguida para atingir uma segmentação de qualidade, executando uma segmentação guiada a partir das estimativas dos objectos. O estudo é finalizado com a integração de informação de movimento nas técnicas mais promissoras.

Como uma avaliação justa de um algoritmo de segmentação requiere uma métrica apropriada, esse trabalho é precedido por um estudo preliminar sobre métricas para comparar segmentações de imagens. O desempenho insuficiente das métricas existentes conduziu a uma investigação exaustiva de novas soluções, culminando na redescoberta das medidas baseadas no grafo-intersecção de duas segmentações, e na sua introdução à comunidade de processamento e análise de imagem pela primeira vez.

Apresentam-se vários casos de estudo para evidenciar a validade das métricas propostas e das técnicas de segmentação de imagens assistidas por metadados.

Palavras-chave: segmentação de imagem, avaliação objectiva da qualidade da segmentação, fusão de dados, metadados, informação de profundidade, informação de movimento

Abstract

For over a century the process of capturing electronic images has remained virtually unchanged, with each pixel in the image being a discrete sample of the spatial and temporal continuum being photographed. In a conventional camera, the only recorded information for each pixel is position and colour. The fact is that captured images remain very limited samples of the scene they represent, only giving a 2D impression of the 3D spatial build-up of the scene. This primitive process of capture is the cause of problems in a myriad of applications, ranging from the film and television post-production to virtual reality, or object-based video formats and industrial inspection. Although much effort has been put into surmounting these problems, all these approaches are based on the estimation of data that is simply not included in the discrete samples provided by digital images, and so are limited in the quality they can provide. The capture of additional data is a step forward to address these problems.

The study of enhanced image segmentation techniques is critical given that image segmentation is an ubiquitous operation, spanning a large set of applications, and that subsequent processes rely heavily on its performance. The availability of additional data that is synchronous with colour information should significantly boost the performance of current state of the art techniques, fostering a whole class of new applications.

This dissertation focuses on the study of image segmentation techniques assisted by meta-data. In particular, the use of depth and motion information to improve the quality of segmentations of colour images is investigated. Starting from primitive fusion approaches for colour and depth, the work evolves to richer techniques, with the joint modelling of colour and depth information; the the method is further extended to conveniently handle the noise typical of real-life depth data. Next, an alternative approach is presented, where depth is used for providing a crude identification of the objects in the image and colour is then used to attain high-quality borders, performing a guided image segmentation starting from the crude seeds obtained from the depth information. The study is concluded with the integration of motion information in the most promising fusion techniques.

Because a fair judgment of any new image segmentation algorithm needs a fair comparison metric, a preliminary study on metrics for comparing image segmentations was conducted in the first place. The poor performance of existing measures led to an exhaustive investigation on new solutions, culminating on the rediscover of the metrics based on the intersection-graph of two segmentations and on their introduction to the image engineering community for the first time.

In the numerous experiments that are reported, experimental evidence of the adequacy of the metrics and enhanced images segmentation techniques is provided.

Keywords: image segmentation, objective evaluation of segmentation quality, data fusion, metadata, depth information, motion information

Résumé

Jusqu'à aujourd'hui, le processus de capture d'images digitales a toujours été basé sur le même principe, selon lequel chaque élément de l'image est un échantillon discret dans l'espace et dans le temps, de la réalité continue à capturer. Néanmoins, les images digitales ne sont qu'un échantillon très limité de la réalité qu'ils représentent, donnant seulement une impression 2D de la disposition 3D des objets. Ce processus limité de capture est source de problèmes dans d'innombrables applications, de la postproduction cinématographique à la réalité virtuelle, en passant par les formats vidéo basés sur la codification d'objets et par l'inspection industrielle. Bien que de diverses techniques aient été développées pour dépasser ces problèmes, la vérité est qu'elles se basent toutes sur l'estimation de données qui, simplement, n'ont pas été capturées, et donc sont limitées dans la qualité qu'elles peuvent atteindre. La capture d'informations supplémentaires apparaît comme la solution la plus naturelle pour dépasser ces problèmes.

Parce que la segmentation d'une image est une opération récurrente dans les plus divers secteurs, et de qui la performance des opérations subséquentes dépend fortement, son étude se rend obligatoire. L'information de profondeur synchrone avec l'information de la couleur devra permettre d'atteindre des performances significativement supérieures à celles possibles avec les techniques actuelles, ouvrant toute une série de nouvelles applications.

Cette dissertation se concentre sur l'étude de techniques de segmentation d'images assistées par des metadata. En particulier, l'utilisation d'informations de profondeur et de mouvement pour améliorer la qualité des segmentations des images en couleur est investiguée. En partant de techniques primitives de fusion des informations couleur et profondeur, le travail évolue pour des techniques plus riches, avec la modélisation commune de la couleur et de la profondeur; la méthode est prolongée pour traiter de façon rigoureuse le bruit typiquement présent dans les informations de profondeur dans des conditions réelles. Ultérieurement, un abordage alternatif est proposé, dans lequel la profondeur est utilisée pour fournir une estimation grossière des objets dans l'image, et la couleur est utilisée ensuite pour obtenir des frontières de qualité élevée, exécutant une segmentation guidée à partir des estimations des objets extraits préalablement. L'étude continue avec le prolongement des techniques de fusion les plus prometteuses pour incorporer information de mouvement.

Comme une évaluation juste d'un algorithme de segmentation a besoin d'une métrique appropriée, ce travail est précédé par une étude préliminaire sur les métriques pour comparer des segmentations d'images. La performance insuffisante des métriques existantes a conduit à une recherche exhaustive de nouvelles solutions, culminant dans la redécouverte des mesures basées sur la grapho-intersection de deux segmentations, les introduisant pour la première fois à la communauté de traitement et d'analyse d'image.

Plusieurs cas d'étude sont présentés pour rendre évidente la validité des métriques proposées et des techniques de segmentation d'images assistées par des metadata.

Mots-clés: segmentation d'image, évaluation objective de la qualité de la segmentation, fusion de données, metadata, information de profondeur, information de mouvement

Preface

In the beginning... was the MetaVision project. A project proposing an innovative electronic production system to reduce the cost of film production and to allow more artistic flexibility in shooting and film editing. It also provided the enabling technology for the integration of real and virtual images at source quality for film production and in TV studios in the compressed domain. A key feature in the MetaVision system is a depth sensor, which provides valuable metadata to aid many post-production processes. A second enhancement of the capture system is the high temporal resolution sensor, further improving the accuracy and quality of subsequent processes. Some of the longer term objectives of the MetaVision project comprise the investigation of how the metadata generated within the MetaVision system can be applied to improve the efficiency of key operations. This was the starting point of this thesis.

Embracing the challenge triggered by the novel MetaVision system, we investigate the use of depth information to assist image segmentation. Depth, because it starts to make sense even in scenarios different from those put forward by MetaVision; image segmentation, because it has long been recognized as one of the most critical steps for automated analysis and is becoming an increasingly important image processing step for numerous applications. All subsequent interpretation tasks like feature extraction, object recognition, and classification depend largely on the quality of the segmentation output.

The main thrust of this text is thus image segmentation. Our intend has been to present the learned lessons and the main breakthroughs that were accomplished in this four-year journey. I trust that you find reading this text a worthwhile investment.

Acknowledgments This work became possible due to the support of some different people and organizations. In particular, I would like to thank my supervisor, Professor Luís Corte-Real, for his support and guidance over the last four years. Thanks to my best-in-the-world officemate Luís F. Teixeira for his collaboration and true friendship; thanks to *Maria* for her never ending confidence in my ability to finish this research and true love. I would also like to thank INESC Porto for providing the right environment for high-quality research. Finally, I thank FCT (Fundação para Ciência e Tecnologia) for financial support.

Jaime dos Santos Cardoso
January 2006

Contents

Resumo	vii
Abstract	ix
Résumé	xi
Preface	xiii
1 Introduction	1
1.1 Motivation	3
1.2 Landscape of image segmentation algorithms	3
1.2.1 Unsupervised segmentation	4
1.2.2 Supervised segmentation	6
1.3 Working methodology	6
1.3.1 The selection of test images	6
1.3.2 Measures of performance	10
1.3.3 Quality assessment needs a reference	10
1.4 Thesis' structure	20
1.5 Contributions	21
2 A unifying model for the evaluation of image segmentations	23
2.1 Evaluation methods for image segmentation	24

2.2	On the discrepancy methods — a review	25
2.2.1	Berkeley measures	27
2.3	A general framework for the comparison of image segmentations	28
2.3.1	The intersection-graph	29
2.4	Discussion	32
3	Partition-distances	33
3.1	The partition-distance, d_{sym}	33
3.1.1	Properties of the partition-distance, d_{sym}	34
3.1.2	Distance d_{sym} applied to binary partitions	36
3.1.3	Efficient computation and graph interpretation of d_{sym}	38
3.2	Asymmetric partition-distance, d_{asy}	38
3.2.1	Efficient computation of d_{asy}	39
3.3	The mutual partition-distance, d_{mut}	40
3.3.1	Graph interpretation of the mutual partition-distance	41
3.3.1.1	Properties of the mutual partition-distance, d_{mut}	42
3.3.2	Connection to the partition-distance	43
3.3.3	Mutual partition-distance as an optimization problem	45
3.3.3.1	Reformulation with a compact convex domain	46
3.3.3.2	Reformulation as a generalization of the partition-distance	47
3.4	Proposed discrepancy measures	48
3.5	Experiments	48
3.6	Discussion	52
4	Data in, data out fusion approaches for hybrid image segmentation	55
4.1	Multisensor information fusion	56
4.1.1	Fusion classification	57

4.2	Intensity substitution approach for hybrid image segmentation	59
4.2.1	Results	60
4.3	Multiresolution approach for hybrid image segmentation	62
4.3.1	Results	62
4.4	Discussion	64
5	Data concatenation approach for hybrid image segmentation	65
5.1	Contour refinement	69
5.2	Discussion	71
6	Hybrid image segmentation by fusion of decisions	73
6.1	Automatic marker extraction	74
6.1.1	Marker refinement	75
6.2	Guided image segmentation	78
6.3	Discussion	82
7	Image segmentation assisted by depth and motion information	83
7.1	Motion and depth assisted image segmentation with the Mean Shift algorithm	85
7.2	Image segmentation guided by noisy metadata	86
7.3	Results with synthetic material	91
7.4	Results with real-life material	93
7.4.1	The ‘Outdoor’ sequence	93
7.4.2	The ‘Indoor’ sequence	97
7.4.3	Image sequence processing	98
7.5	Discussion	102
8	Conclusion	103
	References	105

Chapter 1

Introduction

When we consider what is happening when an electronic image is captured, we note that each pixel in that image is a discrete sample of the spatial and temporal continuum being photographed. It is a sample in spatial terms (it describes a specific x, y position, or more exactly, a particular direction of view) but it also represents a temporal sample. In fact, we can say that the pixel point contains spatial position, direction of view, illumination level, colour sample, temporal integration over a given shutter time, depth in the scene represented by the pixel and direction of motion of the pixel content with time. In a conventional camera, the only recorded information for each pixel is position and colour values. The fact is that captured images remain very limited samples of the scene they represent. The two-dimensional image only gives an impression of the spatial build-up of the scene. [1]

The problems begin when, for example, the frame-rate of the presentation requires intermediate images, or when an element needs to be added to the image in a seemingly correct location in 3D space. Among the many growing applications fields demanding such operations are film and television post-production, object-based video formats, human computer interaction, industrial inspection and virtual reality, to name just a few.

Much effort has been put into surmounting these problems. Interpolation algorithms seek to fill in the missing temporal information. 3D information can be recovered and continuously refined, using information cues such as structure-from-motion or structure-from-shading. Nevertheless, all these approaches are based on the estimation of data that is simply not included in the discrete samples provided by digital images, and so are limited in the quality they can achieve. [1]

The capture of additional data is a step forward to address these problems. The European project MetaVision [2] carried out a survey into possible methods for capturing depth information, details of which can be found in [3,4]. Time-of-flight principle cameras are now

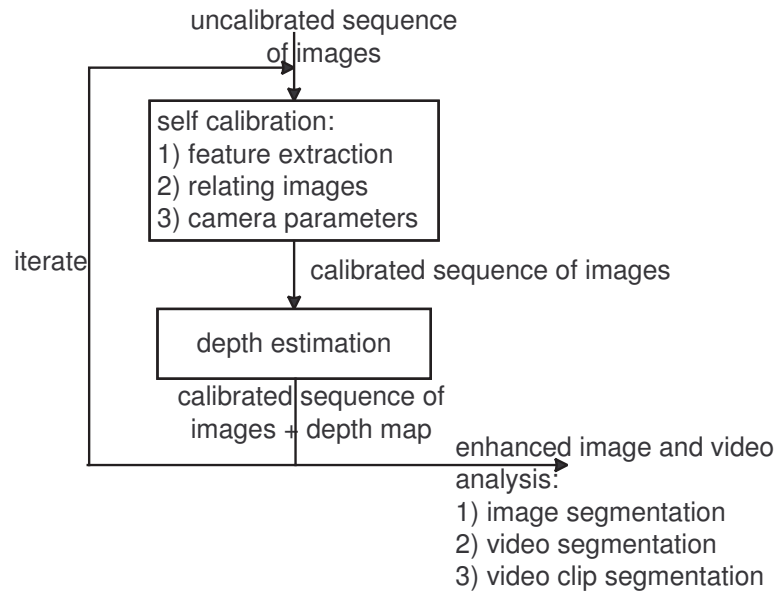


Figure 1.1: System overview for enhanced operations from simple colour sequences.

becoming available (see for example [5–8]). These recent low-cost 3D sensors are capable of real-time 3D acquisition. European projects, like MetaVision [9] have also investigated how to capture and record additional data and how to use it to assist the post-production workflow. Multimodal techniques based on the joint processing of colour and depth data are going to gain momentum.

The capture of additional information is not the only scenario where the processing of depth data synchronously with colour information is an opportunity. Another stimulus comes from the availability of techniques that build up and continuously refine a 3D model of a scene using information cues such as structure-from-motion, structure-from-shading, stereo information, etc [10]. These model based techniques are particularly suited when the physical objects in the 3D scene represent the content we are interested in. Model-based analysis of video sequences has been shown to be a promising approach for various applications like video coding, object tracking and object recognition. Under these techniques an initial model is extracted from the first frames of the sequence and is continuously refined and updated along the image sequence. The recovered 3D motion and depth information allows to perform enhanced operations over the sequence such as to manipulate and separate objects in the scene, image and video segmentation, etc (see figure 1.1).

1.1 Motivation

The availability of the additional depth data captured synchronously with colour information gives rise to new and exciting areas of research. Besides the already referred, compression video formats are also among the applications that can, potentially, benefit from this depth data. Object based video formats, as MPEG-4, are the obvious answer. We could use the depth data to improve the segmentation of images to enable low bit-rate coding using MPEG-4: a low-resolution depth signal would be sufficient to identify parts of the scene that should be encoded with higher quality, or that would be appropriate for encoding using tools such as sprites. In another scenario, depth information could be used to select the important areas of a scene that will be represented as arbitrary-shaped video objects, with the remainder of the scene being represented as, for example, a sprite object. Camera motion data could be used to control the sprite position. This method of encoding could be compared with non-object-based encoding to see whether there is an improvement in coding efficiency. Although similar tests have been carried out in the development of the MPEG-4 standard, it is worth noting that the required shape signals have often been derived by a process requiring a lot of manual intervention [2].

We aim at investigating the use of depth data to allow the development of better image segmentation algorithms that might be able to enhance subsequent operations. Note that depth information alone is insufficient for general object segmentation, since it is typically too noisy to provide a useful segmentation; moreover objects of interest will often be on the ground, and any simple segmentation technique applied to the depth signal will inevitably select regions of ground as well. When processing a sequence of images, the computed segmentation can be improved by exploiting the temporal dependence of consecutive frames to derive additional metadata in the form of motion information.

1.2 Landscape of image segmentation algorithms[§]

Image segmentation is the first important process in innumerable applications, with subsequent processes relying heavily on its performance. It partitions the image into different meaningful regions with homogeneous characteristics using discontinuities or similarities of image components. In most cases, the segmentation of colour images demonstrated to be more useful than the segmentation of monochrome images, because colour images expresses much more image features than monochrome images.

According to the usage of prior knowledge of the image, colour images can be segmented in an *unsupervised* or *supervised* way. The former attempts to construct the “natural

[§]The following introduction to image segmentation is based largely on [11,12].

grouping” of the image without using any prior knowledge. The latter, however, separates the image based on the sample of the object colours. The unsupervised segmentation is widely used in the applications where the image features are unknown, such as natural scene understanding, satellite image analysis, etc. The supervised segmentation is commonly used in the applications where the sample of the object colours can be acquired in advance, e. g., object tracking, face/gesture recognition, and image retrieval, etc.

1.2.1 Unsupervised segmentation

The spatial compactness and colour homogeneity are two desirable properties in unsupervised segmentation, which lead to *image-domain* and *feature-space* based segmentation techniques. According to the strategy of spatial grouping, *image-domain* techniques include *split-and-merge*, *region growing* and *edge detection* techniques. There have been extensive studies on them in the literature. In [13], the Markov random field (MRF) is defined in the quad-tree structure to represent the continuity of colour regions in the process of split-and-merge. In [14], splitting and merging phases are operated by the watershed transform and self-organizing map (SOM), respectively. Zhu developed a segmentation algorithm named as “Region Competition” in [15]. It combines the global optimization methods (snakes/balloons and region growing) to guarantee the convergence of global optima. Manjunath [16] defined the J-image using local windows in a quantized class-map. The high and low values in J-images correspond to possible boundaries and centers of the regions. The minimum vector dispersion (MVD) operator is proposed to reduce the colour vector to a scalar value in [17]. It is a bias free operator for step edges which produces a strong response for true ramp edges. A circular compass operator is proposed to detect colour edges in [18]. The orientation of ‘needle’ with the maximum difference indicates the edge direction, and its magnitude yields a measure of edge strength.

In *feature-space* based techniques, image segmentation is accomplished by exploiting the homogeneous regions in feature space. The common techniques include *histogram thresholding* and *colour clustering*. The histogram thresholding is a technique that seeks the peaks or valleys in 3 colour histograms or a three-dimensional (3D) histogram. The HSV histograms are used for the segmentation of colour image in [19]. The achromatic regions are determined by the saturation values, and the remaining chromatic regions are segmented by thresholding the peaks of hue histogram. A 3D colour histogram is built by $L^*u^*v^*$ colour components in [20]. The valleys of colour histogram are identified by the watershed algorithm.

The nonparametric clustering is a promising solution in colour clustering. The standard techniques can be categorized as hierarchical or partitional clustering [21]. In hierarchical clustering, only local neighbours involve the cluster merging/splitting by the form of dendrograms. The global knowledge of clusters is not incorporated in the procedure of

clustering. The partitional clustering, on the other hand, is an iterative procedure that directly decomposes the data set into a number of disjoint clusters by minimizing the criterion function (e.g., sum-of-squared-error). The k-means and ISODATA are well-known techniques of partitional clustering. However, they suffer the problems of local optima, clustering reproducibility and initialization sensitivity. The k-mean and ISODATA clustering require the number of clusters to be known *a priori*. In order to determine the optimal number of clusters, Turi [22] proposed a validity measure using the ratio of intra-cluster and inter-cluster measures incorporated with a Gaussian multiplier. The optimal number of clusters is found by minimizing the validity measure.

Some new techniques have been proposed for colour clustering in the literature. Comaniciu employed the mean shift analysis for the exact estimation of clustering kernel in [23]. The spheres with the predefined size are used to search the centers of colour clusters in colour space. It has shown the good performance on segmenting the images with strong variations of density. An interesting category of algorithms originate from graph theory. These methods use the Gestalt principles of perceptual grouping to form the image regions. In general, these methods represent the relations between image entities using graph structures and several related algorithms have been proposed [24–26]. The graph theoretic methods introduce ideas from perceptual grouping to the field of computer vision. The image plane is represented by a graph, the nodes of which correspond to the image entities, and the links convey the relations between these entities. Associated with each graph link (or edge) there is a weight indicating the (dis)-similarity of the two pixels (or regions). The graph is usually represented using the adjacency, or the Laplacian matrix. These algorithms try to divide the initial graph into subgraphs that correspond to image regions [27]. Several methods of this category are based on the notion of graph cuts that are derived from the spectrum of the graph. The spectrum comprises of the eigenvalues and eigenvectors of the matrix representation [27]. Another group of methods is based on agglomeration heuristics to form the final subgraphs based on merging or splitting operations [24]. Grady [26] introduced an alternate idea that finds partitions with a small isoperimetric constant, requiring solution to a linear system rather than an eigenvector problem. Graph segmentation algorithms regularly base their operation on a locally computed pairwise dissimilarity measure that is used to determine the link weights. These weights are supposed to take into account some of the basic factors of visual grouping and their selection is critical for the final segmentation result. Usually, weights are extracted locally using feature distance criteria and region-merging operations are also performed on a local scale that, unless guided by some form of global image information, can lead to suboptimal solutions and erroneous segmentation results.

1.2.2 Supervised segmentation

In supervised segmentation, the pixel classifier is trained for the best partition of colour space using the sample of object colours. The image is segmented by assigning the pixel to one of the predefined classes. The common techniques of supervised segmentation are evaluated in [28], including maximum likelihood, decision tree, nearest neighbour and neural networks. The supervised segmentation is employed for the segmentation of video shots in [29]. The segmentation of image frames is hierarchized by three classifiers, i.e., k nearest neighbour, naïve bayes, and support vector machine. In [30], image segmentation is performed by a procedure of supervised pixel classification. The rule of minimum distance decision is used to assign each pixel to a specific class in a colour texture space. A different sort of supervised segmentation is the semi-automatic approach, where a so-called hint image is used to guide a colour based process. Manually-drawn hint images are already used by some commercially-available tools in applications such as Adobe Photoshop. A review of such algorithms can be found in [28]. Curve evolution techniques [31–33] are quite adapted for this sort of application. Active contours (or snakes) are curves defined within an image domain that can move under the influence of internal forces coming from within the curve itself and external forces computed from the image data. The internal and external forces are defined so that the snake will conform to an object boundary or other desired features within an image.

1.3 Working methodology

There are various types of methodologies that we can use to approach a research. Choosing the right methodology is a crucial step to helping you attain your goal, a valid scientific study. We have decided that the following steps were important to the work being developed.

1.3.1 The selection of test images

The type of image or objects present within the image used to assess the quality of a segmentation method has a strong influence upon the results. Obviously, if the objects of the scene are spatially homogeneous, any sound method will provide good results. In this case, the benefit of the fusion is questionable since standard methods will lead to satisfactory results.

The lack of standard and difficult cases for the evaluation of the joint use of colour and depth information led us to select our own test image set. Although the depth information is typically noisier and with lower bandwidth than colour information in real systems, our study begins with synthetic images with accurate depth maps, as depicted in figure 1.3. Images ‘chess’, ‘billiards’ and ‘teacup’ were rendered with V-Ray 1.45.70 using 3dsmax 6.0;



Figure 1.2: The MetaVision camera (from [3, 4]).

the depth maps were generated using 3dsmax's scanline renderer.[†] Although the size of each of these images is 720×540 , and because the signature of the author could distort the analysis, only the first 520 lines were effectively used for testing, by cropping each image to 720×520 . The size of 'cones' image is 450×375 , being available from the Middlebury Stereo Vision Page[‡]. Although this is not an image with perfect depth information, its quality was considered high enough to group it here.

This methodology allows discarding bad fusion practices, with the confidence that their poor performance is not due to noisy depth information. However, because we aim at achieving algorithms not relying on the accuracy of the depth information — that would hinder the application of such techniques in the real-world — we carry on our study with real images and depth maps.

The MetaVision system captures depth information under the form of two additional image streams from two auxiliary cameras placed either side of the main camera, figure 1.2. The auxiliary cameras are small RGB cameras with normal TV resolution (704×576), whereas the main camera has HDTV resolution (1920×1080). The project implemented a spatially-recursive algorithm based on ideas developed for motion estimation [34], operating on rectified images (details can be found in [3, 4]). Using the MetaVision system, several sequences were captured and the disparity estimated. Four images from such material are going to be used to gauge the performance of algorithms in the presence of noisy depth information, see figure 1.4.

[†]These images are used with permission of the author.
They are available at <http://www.richardrosenman.com/dofpro-cgsamples.htm>
[‡]<http://cat.middlebury.edu/stereo/newdata.html>



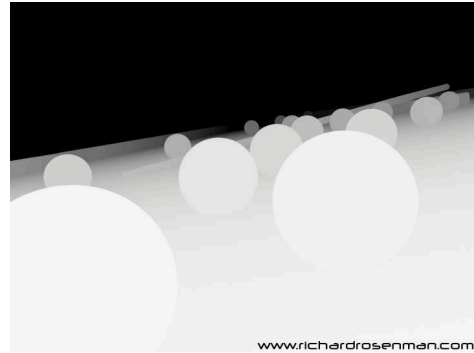
(a) 'chess' image.



(b) 'chess' depth map.



(c) 'billiard' image.



(d) 'billiard' depth map.



(e) 'teacup' image.



(f) 'teacup' depth map.



(g) 'cones' image.



(h) 'cones' depth map.

Figure 1.3: Image test set with perfect depth information.



(a) ‘walk street’ image.



(b) ‘walk park’ depth map.



(c) ‘walk park’ image.



(d) ‘walk park’ depth map.



(e) ‘juggler’ image.



(f) ‘juggler’ depth map.



(g) ‘men’ image.



(h) ‘men’ depth map.

Figure 1.4: Image test set with noisy depth information.

1.3.2 Measures of performance

When in possession of an image segmentation algorithm, an obvious question is “how good is it?”. This begs the question of what we mean by good. The obvious answer is to subjectively judge the result of the application of the algorithm over the pre-defined test set. Although the visual assessment of the segmentations will not be relinquished, the comparison of the methods will rely heavily on the quantitative measures to be introduced in the first part of this work, namely d_{sym} and d_{mut} .

The partition-distance d_{sym} is a strict discrepancy measure between two segmentations of the same image; under- and over-segmentations are appropriately penalized. This measure attains the zero value only when the two segmentations coincide exactly. However, for some purposes, it is important to have measures tolerant to mutual refinements [35], relaxing the conditions for proximity between two segmentations. Intuitively, two segmentations are consistent if they are partially a over-segmentation, partially a under-segmentation of each other. This consistency is effectively measured with the mutual partition-distance, d_{mut} . These measures have the advantage of producing a error mask image, which may assist the evaluation process and the identification of the main source of errors. Besides using the numerical values, we will also make use of the error mask image for d_{mut} ; see the next chapters for details on these measures.

1.3.3 Quality assessment needs a reference

With the objective assessment measures, the quality is assessed with respect to the reference segmentation. As such a reference was not available, it had to be created. With the segmentation tool used to produce the Berkeley segmentation database [36], and publicly available[§], reference segmentations were manually created for the test set, shown in figure 1.5.

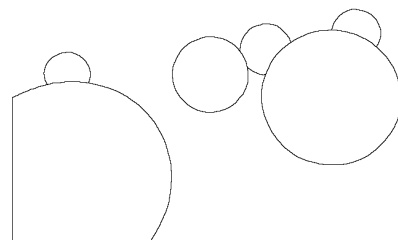
Finally, the multimodal algorithms to be presented in this work had also to be confronted with other algorithms, in order to assess the gain attained with them. As such, the results of the methods to be presented were compared with well established segmentation methods. Three methods were selected. They are all relevant in the community of researchers in image processing.

JSEG algorithm Deng [16] proposed a new approach for colour image segmentation called JSEG which can be used to segment images into homogeneous colour-texture regions. The basic idea of the algorithm is to separate the segmentation process into two independent

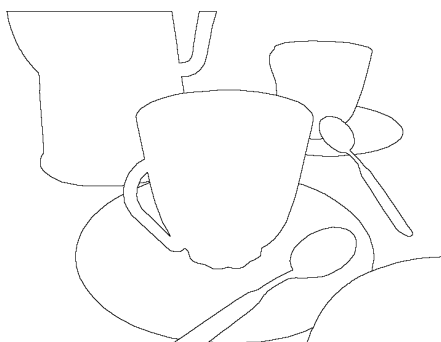
[§]<http://www.cs.berkeley.edu/projects/vision/grouping/segbench/>



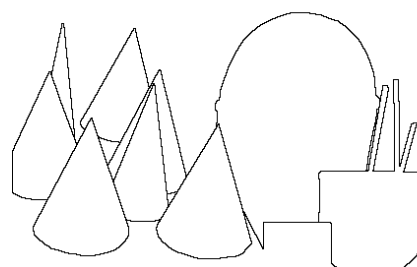
(a) 'chess' image ground truth.



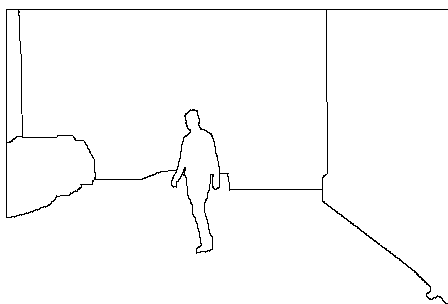
(b) 'billiards' image ground truth.



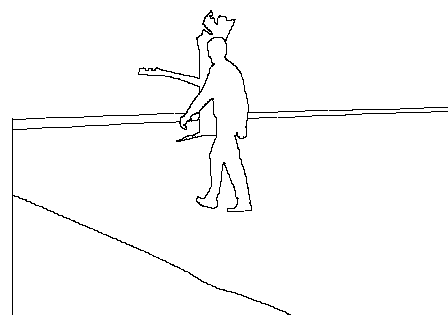
(c) 'teacup' image ground truth.



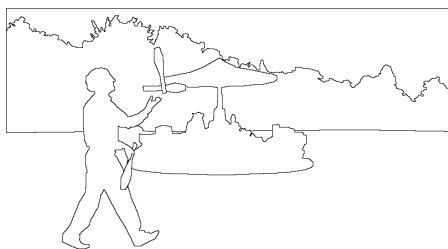
(d) 'cones' image ground truth.



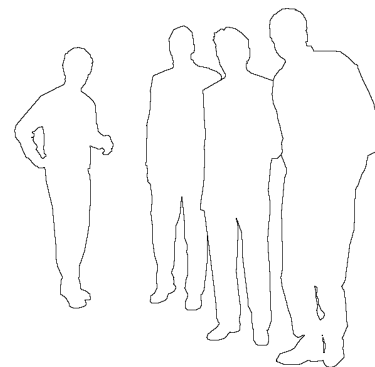
(e) 'walk street' image ground truth.



(f) 'walk park' image ground truth.



(g) 'juggler' image ground truth.



(h) 'men' image.

Figure 1.5: Ground truth segmentations.

stages, colour quantisation and spatial segmentation. In the first stage colours in the image are quantized to several representative classes that can be used to differentiate regions in the image. This quantisation is performed in the colour space *without* considering the spatial distribution of the colours. In the second stage, each pixel's colour is replaced with its class label, thus forming a class-map of the image. The class-map can be viewed as a special kind of texture composition.

Let \wp be the set of all N data points in the class-map. Let $p = (x, y), p \in \wp$, and m be the mean,

$$m = \frac{1}{N} \sum_{p \in \wp} p$$

Suppose \wp is classified into C classes $\wp_i, i = 1, \dots, C$. Let m_i be the mean of the N_i data points of class \wp_i ,

$$m_i = \frac{1}{N_i} \sum_{p \in \wp_i} p$$

Let

$$S_T = \sum_{p \in \wp} \|p - m\|^2 \quad \text{and} \quad S_W = \sum_{i=1}^C S_i = \sum_{i=1}^C \sum_{p \in \wp_i} \|p - m_i\|^2$$

S_W is the total variance of points belonging to the same class. Define

$$J = \frac{S_B}{S_W} = \frac{S_T - S_W}{S_W}$$

The J value measures the distances between different classes S_B over the distances between the members within each class S_W . For the case of an image consisting of several homogeneous regions, the colour classes are more separated from each other and the value of J is large. Applying the criterion to local windows in the class-map results in the J -images, in which high and low values correspond to possible region boundaries and regions centers, respectively. A region growing mechanism is then used to segment the image based on the J -images. The algorithm starts the segmentation of the image at a coarse initial scale. Then, it repeats the same process on the newly segmented regions at the finer scale. Region growing often results in over-segmentation. Therefore, these regions are merged based on their colour similarity in the perceptually uniform CIE $L^*u^*v^*$ colour space. JSEG has been successfully applied in a variety of domains [37–39] and modified for improved performance [40].

Mean shift algorithm Image segmentation exploiting the mean shift procedure was introduced in [23]. The mean shift procedure itself, a nonparametric procedure for the analysis of multimodal data, was proposed by Fukunaga [41] but largely forgotten until Cheng's paper [42] rekindled interest in it. After a proper normalization with h_s and h_c , global parameters in the spatial and colour domains, the location and colour vectors are

concatenated to obtain a spatial-colour vector of dimension $d = 2 + 3$ for colour images (or $d = 2 + 1$ in the grey level case). The mean shift procedure is then applied in the combined spatial-colour domain. Each data point becomes associated with a point of convergence which represents the local mode of the density in the d -dimensional space. Convergence points sufficiently close in the joint domain are fused to obtain the homogeneous regions in the image. Finally spatial regions smaller than a predefined value are eliminated. The mean shift has also found application in diverse areas [43, 44].

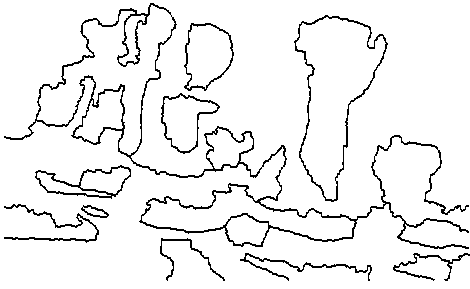
NCut algorithm Normalized Cuts is an unsupervised segmentation technique developed by Shi and Malik [25] that approaches the segmentation problem as a graph-partition problem, using a global criterion. The normalized cut criterion measures both the total dissimilarity between the different groups as well as the total similarity within the groups. The major steps of the NCut algorithm are the conversion of the data to an weighted graph representation using an application appropriate weighting function and transformation of the data clustering problem to a graph partition problem, partitioning of the weighted graph by rewriting the normalized cut objective function as an eigenproblem, solution of this eigenproblem and finding of the Fiedler vector (and possibly other eigenvectors), and finally separation of the data into segments corresponding to clusters in the eigenvector(s) which reveal the data’s features of interest. NCut has also generalized to many application domains [45, 46].

The results of applying these three algorithms to the test set are presented in figures 1.6, 1.7, 1.8, 1.9, 1.10 and 1.11, and tables 1.1 and 1.2.

It is important to observe that the ‘chess’ image is a particularly difficult image, which is exposed in the low quality segmentations of conventional methods. The ‘walk street’ and ‘walk park’ images, being mainly a background / foreground segmentation problem reveal the problem of consistently identify the man as a whole, being the head always combined with the background. Moreover, the tendency of these algorithms to over-segment an image is also reflected in the results.

		‘chess’	‘billiards’	‘teacup’	‘cones’
JSEG	regions	21	44	35	101
	d_{sym}	44.96	62.73	26.71	55.02
	d_{mut}	24.77	3.32	3.25	3.21
Mean Shift	regions	54	41	55	49
	d_{sym}	48.77	56.24	65.18	69.85
	d_{mut}	10.40	1.05	16.58	6.19
NCut	regions	40	40	40	40
	d_{sym}	82.78	82.16	72.99	74.88
	d_{mut}	17.66	3.73	9.68	12.18

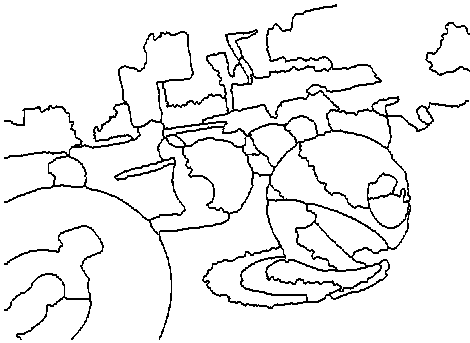
Table 1.1: Results for conventional algorithms, over the test image set with perfect depth information.



(a) 'chess' image.



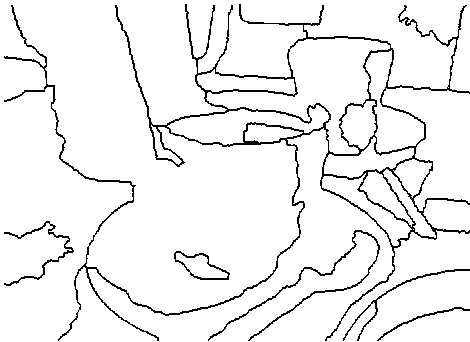
(b) 'chess' error mask.



(c) 'billiards' image.



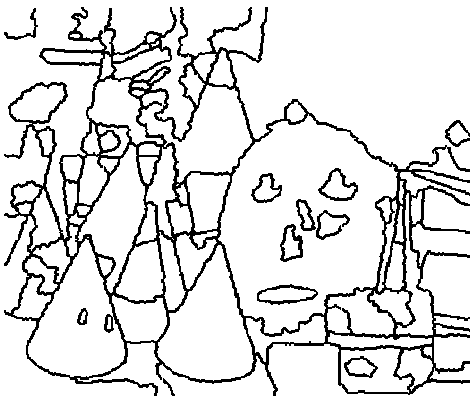
(d) 'billiards' error mask.



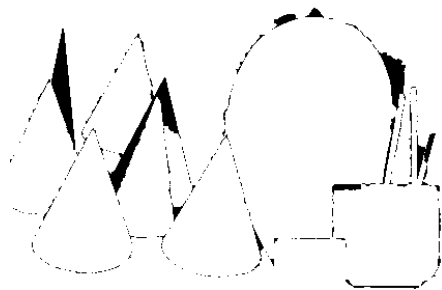
(e) 'teacup' image.



(f) 'teacup' error mask.

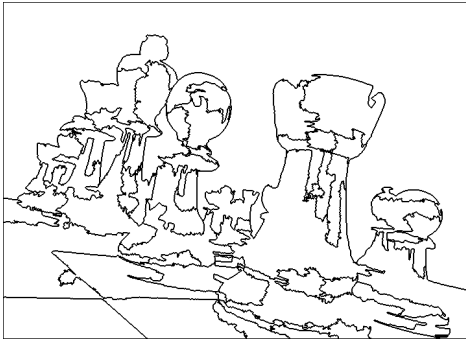


(g) 'cones' image.



(h) 'cones' error mask.

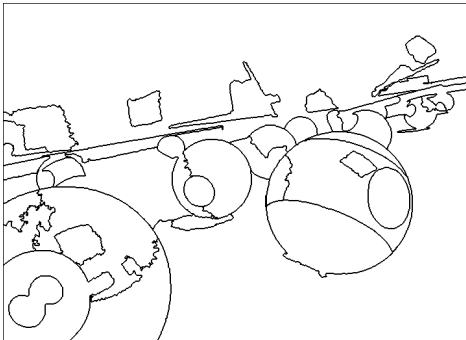
Figure 1.6: Results for the JSEG algorithm over the test dataset with perfect depth information.



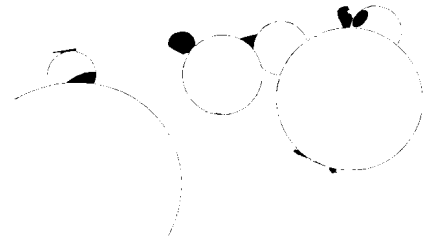
(a) 'chess' image.



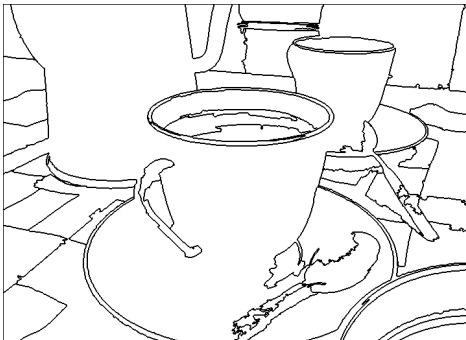
(b) 'chess' error mask.



(c) 'billiards' image.



(d) 'billiards' error mask.



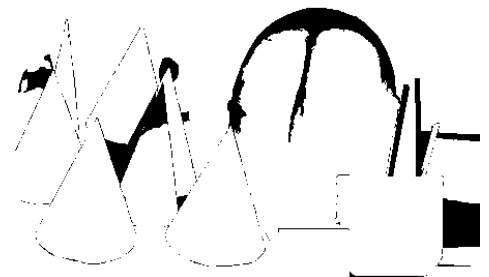
(e) 'teacup' image.



(f) 'teacup' error mask.

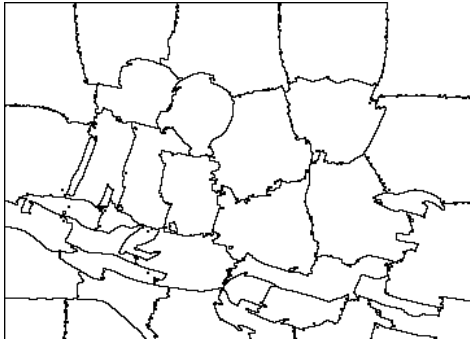


(g) 'cones' image.



(h) 'cones' error mask.

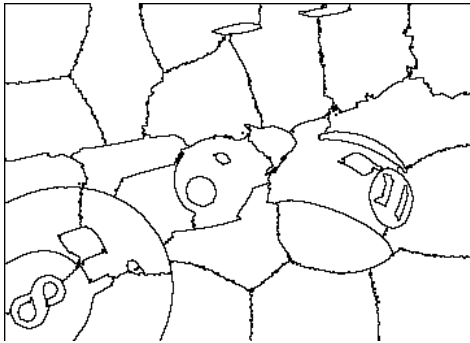
Figure 1.7: Results for the Mean Shift algorithm over the test dataset with perfect depth information.



(a) 'chess' image.



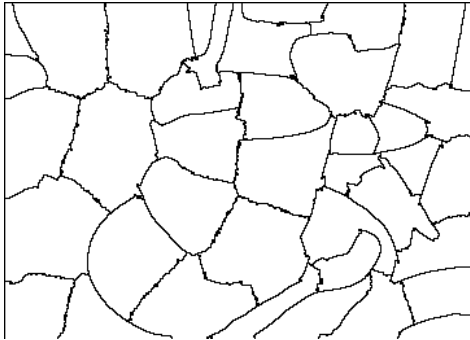
(b) 'chess' error mask.



(c) 'billiards' image.



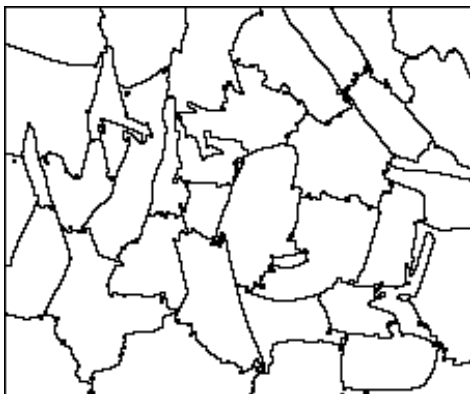
(d) 'billiards' error mask.



(e) 'teacup' image.



(f) 'teacup' error mask.

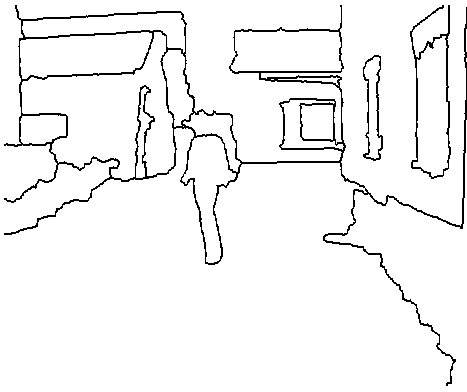


(g) 'cones' image.

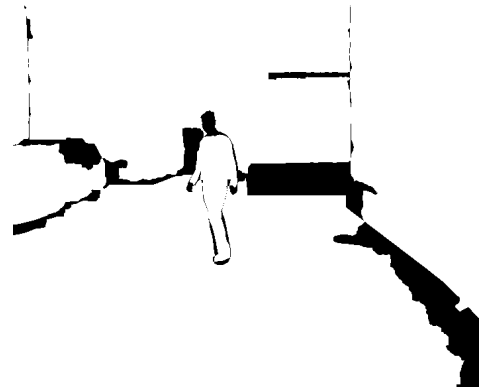


(h) 'cones' error mask.

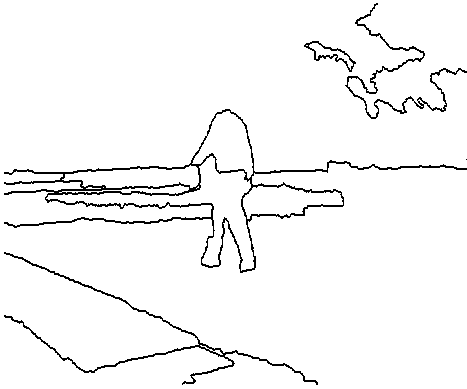
Figure 1.8: Results for the NCut algorithm over the test dataset with perfect depth information.



(a) 'walk street' image.



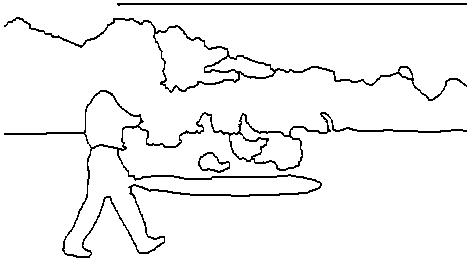
(b) 'walk street' error mask.



(c) 'walk park' image.



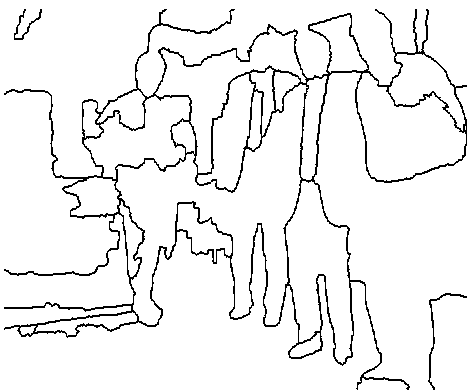
(d) 'walk park' error mask.



(e) 'juggler' image.



(f) 'juggler' error mask.



(g) 'men' image.

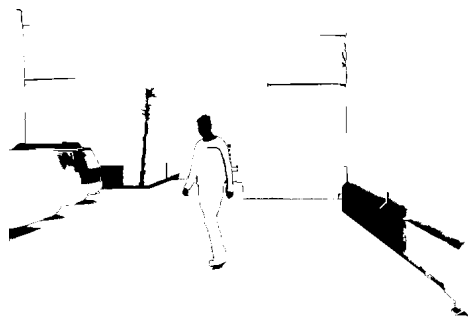


(h) 'men' error mask.

Figure 1.9: Results for the JSEG algorithm over the test dataset with noisy depth information.



(a) 'walk street' image.



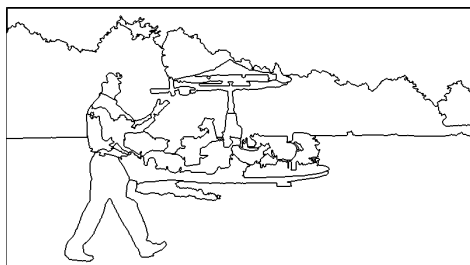
(b) 'walk street' error mask.



(c) 'walk park' image.



(d) 'walk park' error mask.



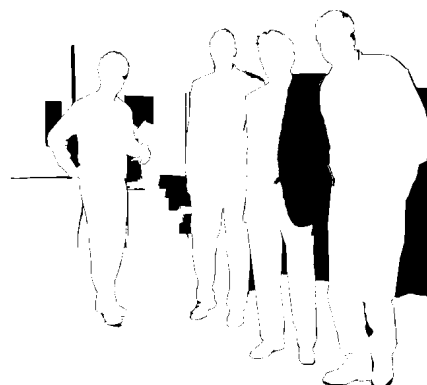
(e) 'juggler' image.



(f) 'juggler' error mask.

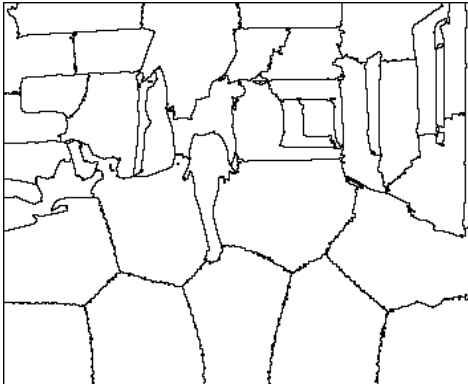


(g) 'men' image.



(h) 'men' error mask.

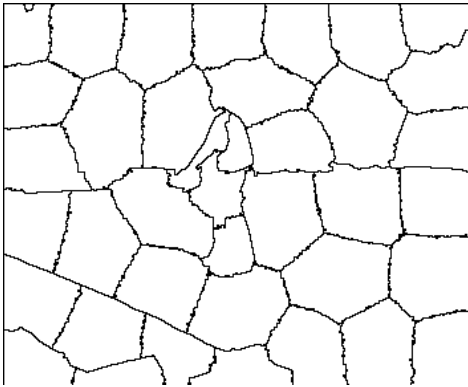
Figure 1.10: Results for the Mean Shift algorithm over the test dataset with noisy depth information.



(a) 'walk street' image.



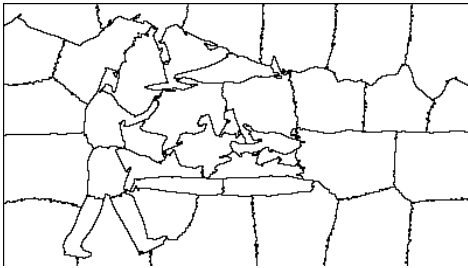
(b) 'walk street' error mask.



(c) 'walk park' image.



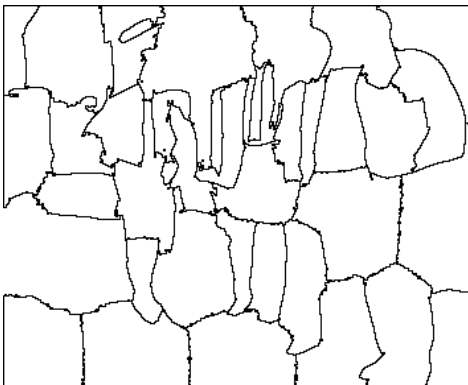
(d) 'walk park' error mask.



(e) 'juggler' image.



(f) 'juggler' error mask.



(g) 'men' image.



(h) 'men' error mask.

Figure 1.11: Results for the NCut algorithm over the test dataset with noisy depth information.

		‘walk street’	‘walk park’	‘juggler’	‘men’
JSEG	regions	22	13	12	35
	d_{sym}	34.90	19.59	16.76	64.26
	d_{mut}	7.91	4.87	10.11	18.11
Mean Shift	regions	40	49	26	50
	d_{sym}	41.50	29.78	25.54	66.12
	d_{mut}	4.20	2.92	4.32	9.46
NCut	regions	40	40	40	40
	d_{sym}	81.4	87.56	83.51	83.30
	d_{mut}	9.09	5.65	5.71	14.59

Table 1.2: Results for conventional algorithms, over the test image set with noisy depth information.

1.4 Thesis’ structure

This thesis introduces in chapter 2 the intersection graph associated with two segmentations of the same image. Starting with a review of the state-of-the-art measures for comparing two segmentations of the same image, the chapter evolves to the definition of the intersection graph as a factory of discrepancy measures; depending on the problem at hand, a measure can be selected that best suites our needs. The most promising measures, constructed from the intersection graph are presented in chapter 3. The partition-distance, a strict measure between two segmentations, ideal for benchmarking, is the first proposed measure. Next, two asymmetric measures, tolerant to over- or under-segmentation are also presented. Finally, the mutual partition-distance, a measure not reacting to mutual refinements, is the last measure derived from the intersection graph. In chapter 4 we investigate simple methods for depth assisted image segmentation. By creating new colour triplets containing information from the original colour and depth images, we apply standard image segmentation algorithms to the fused image. Having concluded of the unsuitability of the naïve attempts, we start in chapter 5 the investigation of more sophisticated methods. We study extensions of conventional, state of the art algorithms to deal with a second image, the depth data, by a joint modelling of colour and depth. Aware of the noisy nature of the depth information in practical problems we proposed refinements for increased robustness. In chapter 6 we analyse another approach for image segmentation assisted by depth data. Now, the depth information is used primarily to estimate the number and position of objects in the image. Then, guided image segmentation, starting from the markers extracted in the previous step, is performed using mostly the colour information. In chapter 7 motion information is integrated in the segmentation process to further improve the segmentation results. Finally, results are discussed, conclusions are drawn and future work is oriented in chapter 8.

1.5 Contributions

We summarize below the contributions of this thesis toward superior and more efficient image segmentation techniques. In this thesis we have

1. introduced in the image engineering community the *intersection-graph* as a factory of measures for comparing segmentations. Presented also the mapping of previously proposed methods to this framework;
2. obtained through reasoning from this generic framework three significant measures for the image engineering community: the *partition-distance*, the *asymmetric partition-distance* and the *mutual partition-distance*. Several properties of these measures, and relationships among them were established;
3. proposed an *extension to a conventional segmentation algorithm*, the mean shift, to accept depth information for the segmentation process, while coping effectively with the expected noise in depth data;
4. proposed a *new procedure for image segmentation guided by depth and motion information*. In a first stage the depth information is used to estimate the number and localization of objects in the image; next, a guided image segmentation is performed, using essentially the colour information, starting from the guides created in the first step. The extension of this framework to sequences of images, integrating motion information derived from the temporal correlation of consecutive frames, was also studied.

Publications related to the thesis

[9] P. W. Walland, G. Thomas, M. Koppetz, J. S. Cardoso, T. Erseghe, and F. Hericourt, “The application of intimate metadata in post production,” in *Proceedings of Int. Broadcasting Convention (IBC 2002)*, september 2002.

[47] J. S. Cardoso and L. Corte-Real, “Toward a generic evaluation of image segmentation,” *IEEE Transactions on Image Processing*, vol. 14, pp. 1773–1782, november 2005.

[48] J. S. Cardoso and L. Corte-Real, “A measure for mutual refinements of image segmentations,” *IEEE Transactions on Image Processing*, 2006.

[49] J. S. Cardoso and L. Corte-Real, “Image segmentation guided by depth information,” *submitted to IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2006 (CVPR2006)*, 2006.

Chapter 2

A unifying model for the evaluation of image segmentations[§]

Authors currently working in the field of low-level image segmentation frequently point out the need for a standard quality measure that would allow both the evaluation and comparison of all segmentation procedures available. This need arises from the ill-posedness of the image segmentation problem: for the same image, the optimum segmentation can be different, depending on the application.

A problem is well-posed if the solution exists, the solution is unique and the solution depends continuously on the data. If it fails to satisfy at least one of these criteria the problem is ill-posed (in the sense of Hadamard, [50]). The general idea of solving ill-posed problems is to restore well-posedness of the problem by introducing some constraints, implicit and explicit, to the solution. This is often termed as regularization of the problem and it has close connections to bayesian estimation.

Automatic segmentation is, therefore, a problem without a general solution, at least at the current state-of-the-art. A standard quality measure, if available, could be applied to automatically provide a ranking among different segmentation algorithms or to optimally set the parameters of a given algorithm, under a pre-defined framework.

Several methods have been proposed to evaluate the quality of segmentation algorithms. Next we will present the main ideas underlying these methods.

[§]Some portions of this chapter appeared in [47, 48].

2.1 Evaluation methods for image segmentation

In the often cited article by Zhang [51], evaluation methods are broadly divided into two categories: *analytical methods* and *empirical methods*: “The analytical methods directly examine and assess the segmentation algorithms themselves by analyzing their principles and properties. The empirical methods indirectly judge the segmentation algorithm by applying them to test images and measuring the quality of segmentation results.”

Although using analytical methods to evaluate segmentation algorithms avoids the implementation of these algorithms (and so they do not suffer from influences caused by the arrangement of evaluation experiments as the empirical methods do), they have not received much attention mainly because of the difficulty to compare algorithms solely by analytical studies. The analytical methods in the literature work only with some particular models or properties, see Liedtke [3] and Abdou [4].

Empirical methods are further classified into two types: *goodness methods* and *discrepancy methods*.

In the empirical goodness methods some desirable properties of segmented images, often established according to human intuition, about what conditions should be satisfied by an ‘ideal segmentation’, are measured by goodness parameters. The performance of the segmentation algorithms under study is judged by the values of goodness measures. These methods evaluate and rate different algorithms by simply computing some chosen goodness measure based on the segmented image, without requiring the *a priori* knowledge of the reference segmentation. Different types of goodness measures have been proposed. Colour uniformity [52], entropy [53], intra-region uniformity [54, 55], inter-region contrast [56, 57], region shape [58], etc, are some of the measures that have been proposed in the literature.

Empirical discrepancy methods are based on the availability of a *reference segmentation*, also called *gold standard* or *ground truth*. The disparity between an actually segmented image and a correctly/ideally segmented image (the gold standard, which is the best expected result) can be used to assess the algorithm’s performance. Both images (actually segmented and reference) are obtained from the same input image. The methods in this group take the difference (measured by various discrepancy parameters) between the actually segmented image and the reference one into account, i.e., these methods try to determine how far the actually segmented image is from the reference image. In section 2.2 we will cover the early proposed methods in this group.

The distinction between empirical discrepancy methods and empirical goodness methods is not so clear cut when we think about the real meaning of selecting a goodness method with the corresponding goodness parameter(s). There is (at least) one segmentation partition

that maximizes the adopted goodness measure — call it implicit gold standard. By choosing an appropriate discrepancy measure for all other possible segmentations — a rather artificial measure —, we can always mimic the goodness method with the implicit discrepancy measure.

So the difference is in how we model the reference segmentation and in what point of view seems most useful, rather than in any intrinsic difference between the methods themselves. Probably a more meaningful name for goodness methods is *empirical with implicit reference*, contrasting with *empirical with explicit reference* that are the so called empirical discrepancy methods.

Although conceptually similar to discrepancy methods, goodness methods have the advantage of being well suited to integrate unsupervised tools — there is no need to feed the method with any data. Also, our perception of a good segmentation might be easier to convey using these methods.

However, goodness methods also have some drawbacks. By first defining what is going to be measured — the goodness parameters —, we can always construct an algorithm that will outperform all the others under the selected evaluation measure. This algorithm would generate the implicit gold standard partition. This may invalidate any assessment at all, this being especially true when similar criteria are used to design the segmentation algorithms as well as to assess their performance — in fact, goodness measures have been used to design segmentation algorithms.

2.2 On the discrepancy methods — a review

Taking a quick snapshot of what have been proposed so far, it is easy to conclude that current discrepancy evaluation methods lack a general and consistent approach.

Yasnoff [59] proposed to take the number of misclassified pixels and their positions into account for computing two measures: the percentage of area misclassified and the pixel distance error. However, this has only been applied to foreground/background segmentation.

A similar approach appears in [60] with Figure of Merit (FOM) for edge detection evaluation. This method, applied to image segmentation, looks at the segmentation process as an edge map extractor, being only suitable for these binary edge map images. It also does not give a good general response [61].

Zhang, in [62] and [63], suggests the use of the so called 'ultimate measurement accuracy': "if the goal of image segmentation is to obtain measurements of object features, the accuracy of these measurements obtained from the segmented images can be used as a quantitative

evaluation criteria”. Mattana [64] and Huo [65] have followed a similar approach. Although this assumption may be valid in the context of image analysis, more and more applications make use of the regions created in the segmentation process, of which the new object-based compressing standards are just an example.

Chalana’s proposal [66] works only for “... a single object from an image”.

Betanzos [67] defines an accuracy measure for images with multiple types of objects. However, it only works when not all types of objects are present in the image. It also has to be able to count the correct and false results separately for each type of object.

Hoover [68] uses a region-based method for assessment. Nevertheless, he does not avoid unintuitive ad hoc measures that involve user defined thresholds. [69] continues the work of [68] using the same performance evaluation method; [70] proposes an adapted version of the same measure.

Roldan [61] has introduced a hybrid measure of empirical discrepancy and empirical goodness. This measure is only intended for the evaluation of low error segmentation results using the binary edge map of a segmentation.

Belaroussi [71] proposes a set of localization measures that can be used on a binary image under the knowledge of a binary reference image to evaluate the quality of the segmented edges. Although it was adapted to segmentation region maps in [72], that was only done with background/foreground segmentations.

Everingham [73], more than defining a new measure, attempts to aggregate fitness functions using the Pareto front. Measures such as ours could be used as fitness functions in the proposed methodology.

Martin in [74], and more thoroughly in [35], proposes a very interesting set of measures. Most of these measures — GCE, LCE and BCE measures — compute the overall distance between two segmentations as the sum of the local inconsistency at each pixel. A novel methodology for judging the quality of a boundary map is also presented. The correspondence procedure, tolerant to small localization errors, resorts to the bipartite matching of “little pieces of boundary, or edgels”. All measures are general enough to work with images with several objects and they all achieve excellent results in the collection of test images. However, their behaviour is not always the expected, as illustrated later — see section 3.1.1. This is probably due to their local definition, making it also difficult to predict the performance for complex segmentations. Because of the significant quality of the measures introduced by Martin, we detail know each of them, presenting their definition.

2.2.1 Berkeley measures

First, define a quantity $E(S_1; S_2; p)$ called the *local refinement error*, which measures the degree to which two segmentations S_1 and S_2 agree at pixel p . Let $R(S; p)$ be the set of pixels in segmentation S which are in the same segment as pixel p ; $|\cdot|$ denotes cardinality and \setminus set difference. Then

$$E(S_1; S_2; p_i) = \frac{|R(S_1, p_i) \setminus R(S_2, p_i)|}{|R(S_1, p_i)|}$$

Five different measures were introduced in [35] with interest for this work:

1. Local Consistency Error, which permits refinement in different directions in different parts of the image:

$$LCE = \frac{1}{N} \sum_i \min \{E(S_1; S_2; p_i); E(S_2; S_1; p_i)\}$$

2. Global Consistency Error, which forces all local refinements to be in the same direction, i.e. from one segmentation to the other:

$$GCE = \frac{1}{N} \min \left\{ \sum_i E(S_1; S_2; p_i), \sum_i E(S_2; S_1; p_i) \right\}$$

3. Bidirectional Consistency Error, penalizing any difference between the segmentations:

$$BCE = \frac{1}{N} \sum_i \max \{E(S_1; S_2; p_i); E(S_2; S_1; p_i)\}$$

4. Mutual Information Distance. The distance between two segmentations is computed as the mutual information between an affinity function. For segmentation 1, define $F_1^{(ij)}$ as 1 when pixels i and j belong to the same segment, and zero otherwise; identically, define $F_2^{(ij)}$ for segmentation 2. Note that $F_1^{(ij)}$ and $F_2^{(ij)}$ are binary valued. Given the joint distribution $p(x; y) = P(F_1 = x; F_2 = y)$, the mutual information is defined as the Kullback-Liebler divergence between the joint and the product of the marginals:

$$I(F_1; F_2) = \int_{x,y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}$$

5. Edgel based measure. Given two segmentation, it corresponds little pieces of boundary, or edgels. This is the only measure that works on boundaries. An edgel is an oriented edge fragment which has an image-plane position (x, y, θ) , and a length equal to 1 pixel. The error is the proportion of edgels that can not be corresponded.

2.3 A general framework for the comparison of image segmentations

As section 2.2 shows, only a few methods actually explore the segments (clusters) obtained from the segmentation process. Most measures are best suited to evaluate edge detection, working directly on the binary image of the regions' contours. Although we can always treat a segmentation as a boundary map, the problem lurks in the simplified use of the edge map, as simply counting the misclassified pixels, on an edge/non-edge basis. But pixels on different sides of an edge are different in the sense that they belong to different regions — that is why it may be more reasonable to use the segmentation partition itself. Realizing this, some authors have introduced 'artificial corrections' to improve measures, notably counting the misclassified pixels and weighting the erred pixels according to their distance to the reference.

Most of previously proposed methods, working directly on the segments suffer from several limitations, ranging from the number of objects in the image (foreground/background segmentation), see [59, 66], to simplifications introduced in order to be able to tackle the problem [67–69]. A clear exception is the work of Martin in [35, 74]. To our knowledge, none of the proposed methods tries to define a reasonable discrepancy measure from the definition of image segmentation.

Image segmentation is traditionally viewed as a process that partitions the entire image region R into n sub regions, $r_1, r_2, r_3, \dots, r_n$, such that:

1. Every pixel belongs to a region — $r_1 \cup r_2 \cup r_3 \cup \dots \cup r_n = R$
2. Every region is spatially connected
3. All regions are disjoint — $r_i \cap r_j = 0, i \neq j$
4. All pixels in a region satisfy a specified similarity predicate — $P(r_i) = true$
5. For any two adjacent regions, r_i and r_j , $P(r_i \cup r_j) = false$, where P is the mentioned similarity predicate

Since an image segmentation is defined as a partition, when comparing the gold standard with the segmentation under evaluation, we are in fact comparing two partitions. So, how to compare two partitions? At the core of the problem are distance metrics, which define the notion of similarity between two partitions. In general terms, having a set of N elements and two different partitions defined on this set makes it possible to compare the two partitions in many ways — no single metric is useful in all circumstances. First, let us define an entity, hereafter called the *intersection-graph*, which enables to define sensible measures for many applications.

2.3.1 The intersection-graph

Before introducing the concept of intersection-graph, some helpful notions and notation are in order. Let S be a set of N elements. A cluster is a *non-empty* subset of S . A partition of S is a set of mutually exclusive clusters, whose union is S . Two partitions P and Q of S are *identical* if and only if every cluster in P is a cluster in Q . A partition P is a *refinement* of a partition R (or P is *finer* than R) if and only if each cluster in P is contained in some cluster of R — see figure 2.1. Note that then, by definition, any partition is a refinement of itself.

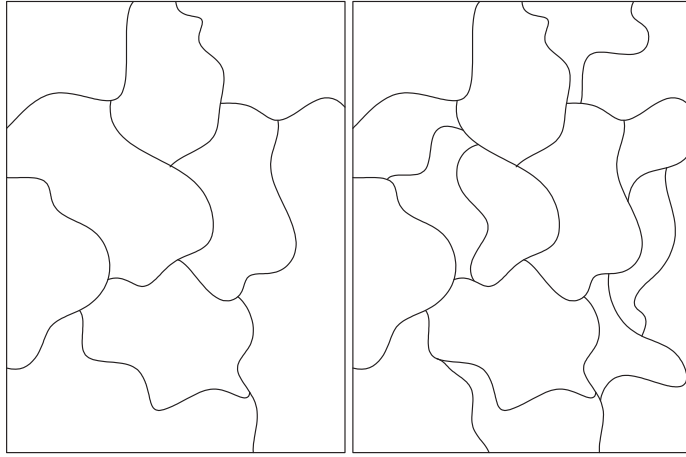


Figure 2.1: The right partition is a refinement of the left partition.

The *intersection of two partitions* P and Q is a partition R so that every non-empty intersection of a cluster S_i from P and a cluster S_j from Q is an element of R — see figure 2.2. Note that R is a refinement of P and Q .

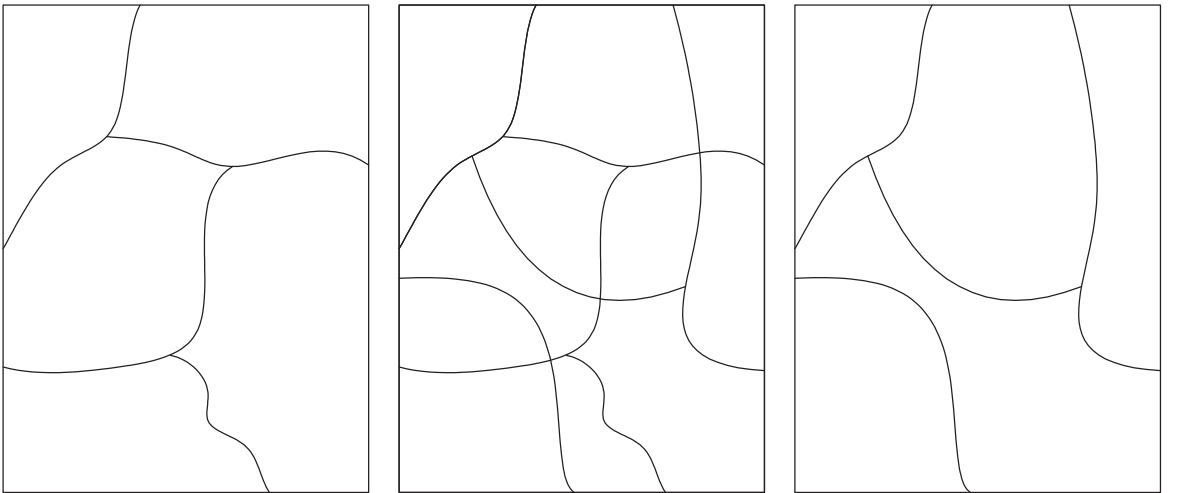


Figure 2.2: The middle partition is the intersection of the left and right partitions.

The *null partition* is the partition with only one cluster (the cluster has N elements). The *infinite partition* is the partition with N clusters (each cluster has one element).

A bipartite graph \mathcal{BG} is a graph whose set of vertices V can be split into two subsets V_R and V_C in such a way that each edge of the graph joins a vertex in V_R and a vertex in V_C — figure 2.3(a). A bipartite graph with r vertices in V_R and c vertices in V_C is denoted by $\mathcal{BG}_{r,c}$.

A complete bipartite graph is a bipartite graph in which each vertex in V_R is joined to each vertex in V_C by an edge — figure 2.3(b). The complete bipartite graph with r vertices in V_R and c vertices in V_C is denoted by $K_{r,c}$.

A tree graph is a simple, undirected, connected, acyclic graph — figure 2.3(c). A tree with n nodes has $n - 1$ edges. Conversely, a connected graph with n nodes and $n - 1$ edges is a tree.

The n -star graph, S_n , is a tree on $n + 1$ nodes with one node having vertex degree n and the others having vertex degree 1. The complete bipartite graph $K_{1,n-1}$ is the star graph S_n — figure 2.3(d).

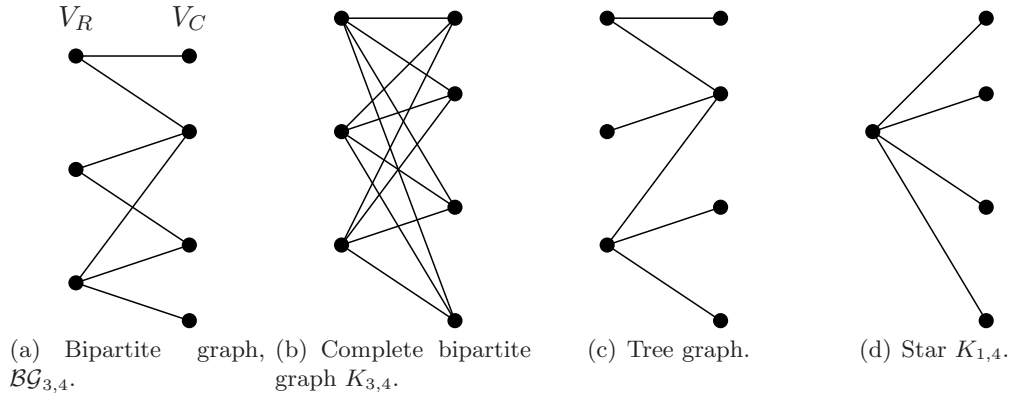


Figure 2.3: Graph definitions.

Given a set S of N elements and two partitions of S , P and Q , define the *intersection-graph* as the bipartite graph $\mathcal{BG}(P, Q)$ with one node in V_R for each cluster in P and one node in V_C for each cluster in Q — see figure 2.4. Connect two nodes x and y by an undirected, weighted edge if and only if x and y intersect each other, assigning to the weight the number of elements in the intersection.

The intersection-graph associated with two image segmentations, as presented previously, can be used as a *factory* of indices of similarity between partitions.

Guigues [75] has already defined a family of symmetric measures on this graph. Although the simplest way is to assign the area of intersection to the weight of each edge, that can be

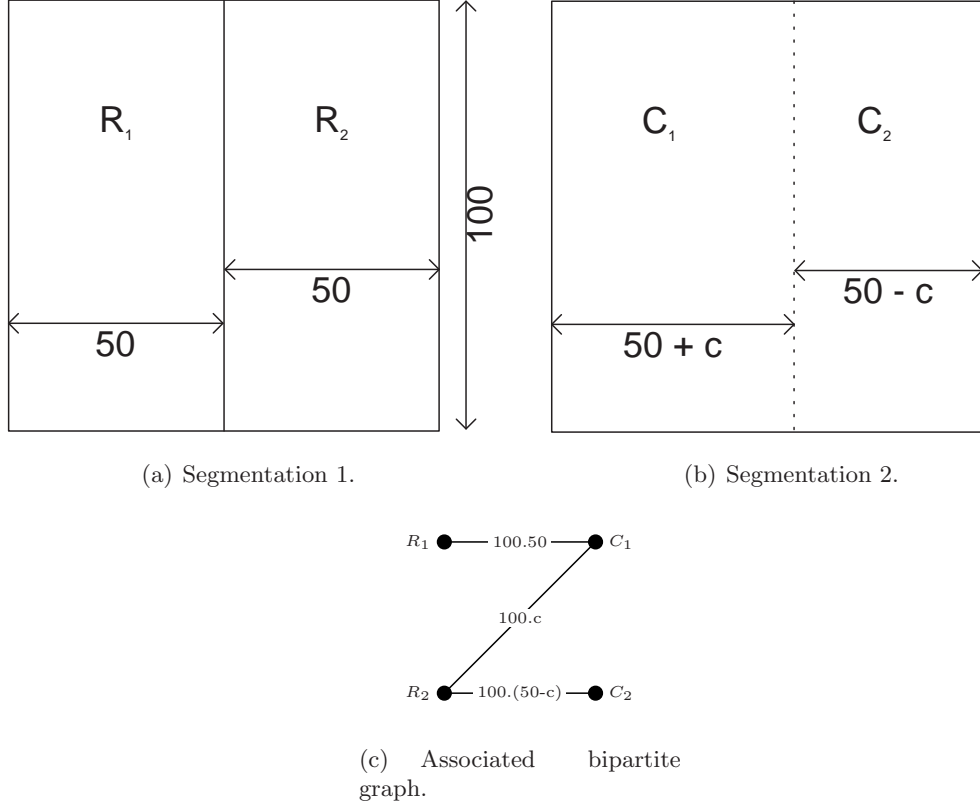


Figure 2.4: Intersection-graph for segmentations 1 and 2.

replaced by any cost function expressing the importance of a region intersection.

More generally, rules can be defined on the vertices and edges of the bigraph to create suitable measures. In fact, many of the previously proposed measures in the literature can be accommodated under this framework. To illustrate, the LCE measure introduced in [35] can be effectively computed as

$$\frac{1}{N} \sum_{\forall \text{ edge } e_i} w_i \cdot \min \left\{ \frac{w_{r_i} - w_i}{w_{r_i}}, \frac{w_{c_i} - w_i}{w_{c_i}} \right\} \quad (2.1)$$

where w_i is the weight of the edge e_i , r_i and c_i are the nodes incident to edge e_i , w_{r_i} is sum of the weights of all the edges incident to node r_i , w_{c_i} is sum of the weights of all the edges incident to node c_i . Latter on in this study we will make use of this general framework to develop a tuned measure for a specific application.

2.4 Discussion

The evaluation of image segmentations is a key element when comparing segmentation algorithms. Segmentation quality evaluation allows the assessment of segmentation algorithms' performance for a given target application and the tuning of algorithms for optimal performance. It is believed that objective evaluation of image segmentation is a very present-day problem, for which a satisfying solution is not yet available in the literature. In this chapter, a general framework for the evaluation of image segmentations, based on the intersection-graph, was presented. While some of the most recent segmentation quality evaluation methods only deal with two objects (foreground and background), metrics defined on the proposed intersection-graph copes with multiple regions in the segmentation partition, using a clean, comprehensive technique. The flexibility of the proposed framework was illustrated by mapping the LCE measure in the intersection-graph.

Chapter 3

Partition-distances[§]

The most promising measures, constructed from the intersection-graph, are now presented. The partition-distance, a strict measure between two segmentations, ideal for benchmarking, is the first proposed measure. Next, two asymmetric measures, tolerant to over- or under-segmentation are also presented. Finally, the mutual partition-distance, a measure not reacting to mutual refinements, is the last measure derived from the intersection-graph.

3.1 The partition-distance, d_{sym}

Having introduced the segmentation evaluation problem as a problem of defining the similarity between two partitions, we can now proceed to the idea of *partition-distance* as it was first presented in [76]. Several alternative (but equivalent) definitions can be given (each more enlightening than the other for some background conditions):

Definition 1: “Given two partitions P and Q of S , the partition-distance is the minimum number of elements that must be deleted from S , so that the two induced partitions (P and Q restricted to the remaining elements) are identical.” [77]

Definition 2: “The partition-distance is equal to the minimum number of elements that must be moved between clusters in P , so that the resulting partition equals Q (by definition, any set that becomes empty is no longer a cluster).” [77]

Proof that definition 1 is equivalent to definition 2:

Let D_1 be the set of $dist_1$ elements given by definition 1 and D_2 be the set of $dist_2$ elements given by definition 2.

[§]Some portions of this chapter appeared in [47, 48].

- a. From definition 1, P equals Q in $S \setminus D_1$. By moving the elements of D_1 in P to the same cluster as in Q , we can set $P = Q$ in S . This implies $dist_2 \leq dist_1$.
- b. From definition 2, P equals Q in the set of unremoved elements $S \setminus D_2$. This implies $dist_1 \leq dist_2$.

From a) and b) we conclude the equivalence of both definitions. \square

From this definition a useful set of properties can be deduced.

3.1.1 Properties of the partition-distance, d_{sym}

Let P, Q, R be partitions defined in a set S of N elements. Then:

1. $d_{sym}(Q, P) \geq 0$
2. $d_{sym}(Q, P) = 0$ if and only if $Q = P$
3. $d_{sym}(Q, P) = d_{sym}(P, Q)$
4. $d_{sym}(Q, P) + d_{sym}(P, R) \geq d_{sym}(Q, R)$
5. $d_{sym}(Q, \text{null partition}) =$
 $= N - (\text{maximal cluster size in } Q)$
6. $d_{sym}(Q, \text{infinite partition}) =$
 $= N - (\text{number of clusters in } Q)$
7. $d_{sym}(\text{null partition}, \text{infinite partition}) =$
 $N - 1 = \text{maximal distance between any two partitions}$
8. the normalized distance, $d_{sym}/(N - 1)$, ranges from 0 to 1
9. let S_1 and S_2 be two disjoint sets. Be P_1 and Q_1 partitions of S_1 , P_2 and Q_2 partitions of S_2 , and $P = P_1 \cup P_2$ and $Q = Q_1 \cup Q_2$ the resulting partitions defined in $S = S_1 \cup S_2$. Then $d_{sym}(P, Q) = d_{sym}(P_1, Q_1) + d_{sym}(P_2, Q_2)$.

Any function with properties 1 to 4 is called a *metric*.

Proof of property 1: Follows directly from definition. \square

Proof of property 2:

- a. If $Q = P$ no points need to be removed from S to make the partitions equal. Then $d_{sym}(Q, P) = 0$.

- b. If $d_{sym}(Q, P) = 0$ the number of points that had to be removed from S to make the partitions equal was 0. That is, the partitions are already equal in S .

□

Proof of property 3: Follows directly from definition 1 of partition distance.

□

Proof of property 4: Let D_1 be the set of $d_{sym}(Q, P)$ elements to be removed in order to equal Q to P ; D_2 be the set of $d_{sym}(P, R)$ elements to be removed in order to equal P to R . Simultaneously, remove from S the elements of D_1 and D_2 (they may have common elements). Then, in the reduced set, we also have $Q = R$. So, removing $d_{sym}(Q, P) + d_{sym}(P, R)$ is enough to make Q and R equal partitions. That implies $d_{sym}(Q, R) \leq d_{sym}(Q, P) + d_{sym}(P, R)$.

□

Proof of property 5: Because two identical partitions have the same number of clusters, we can only keep elements from one cluster of Q in the reduced set. Then, it is easy to see that removing the elements of all clusters of Q , with the exception of those in the biggest cluster, gives the minimum number of elements that need to be removed to equal Q to the null partition.

□

Proof of property 6: Because two identical partitions have the same number of clusters and the same number of elements in each cluster, we can only keep one element from each cluster of Q in the reduced set — otherwise they would belong to different clusters in the infinite partition. It is easy to see that keeping only one element of each cluster of Q (anyone in fact) equals Q to the infinite partition.

□

Proof of property 7: Making $Q = \text{null partition}$ in 6 or $Q = \text{infinite partition}$ in 5 we get the desired equality. Because it is always possible to keep at least one element of S (anyone if fact), $(N - 1)$ is the maximal possible value that d_{sym} can attain.

□

Proof of property 8: By prop 1 and prop 7, $0 \leq d_{sym} \leq N - 1$. Then $0 \leq d_{sym}/(N - 1) \leq 1$.

□

Proof of property 9:

- a. Remove from S_1 $d_{sym}(P_1, Q_1)$ points to make $P_1 = Q_1$ and from S_2 $d_{sym}(P_2, Q_2)$ points to have $P_2 = Q_2$. Then $P_1 \cup P_2 = Q_1 \cup Q_2$ in the set S restricted to the remaining elements. So $d_{sym}(P, Q) \leq d_{sym}(P_1, Q_1) + d_{sym}(P_2, Q_2)$.
- b. Remove from S $d_{sym}(P, Q)$ points to equal P to Q . Be n_1 the points removed from S_1 . Then $P_1 = Q_1$ in S_1 excluded of the n_1 points. Then $d_{sym}(P_1, Q_1) \leq n_1$. In the same way, being n_2 the number of points removed from S_2 , $d_{sym}(P_2, Q_2) \leq n_2$. Then $d_{sym}(P_1, Q_1) + d_{sym}(P_2, Q_2) \leq (n_1 + n_2) = d_{sym}(P, Q)$.

From a) and b) $d_{sym}(P, Q) = d_{sym}(P_1, Q_1) + d_{sym}(P_2, Q_2)$. □

We propose to apply the distance defined above to measure the discrepancy between the reference segmentation (nothing more than a partition of an image) and the segmentation under evaluation. This distance should be applied directly to the segmentation partition (with a different color representing each region) rather than to the edge map.

For instance, consider the two partitions of the same 8×8 image, represented in figure 3.1.

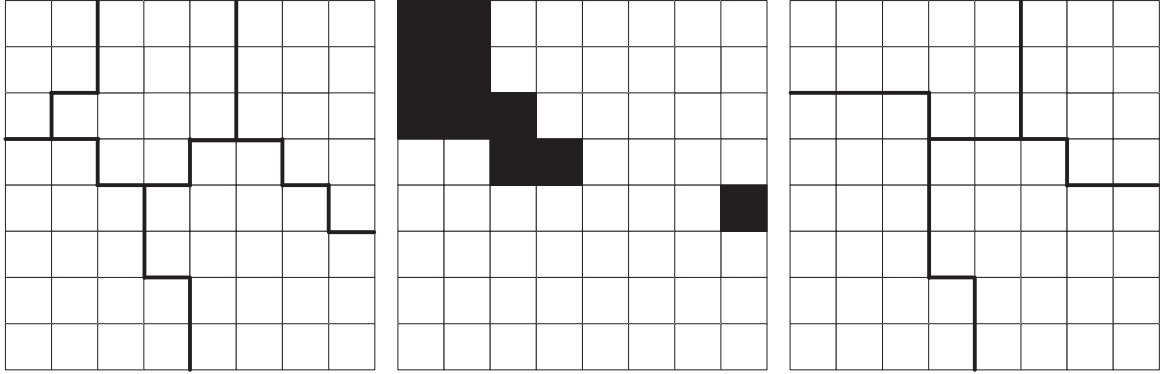


Figure 3.1: Two different partitions of the same image — the middle image highlights the points to be removed.

According to the distance defined above, these partitions are 10 pixels away from each other. The pixels that had to be removed are highlighted in the middle image (unique solution in this particular case). Later, it will be shown how to efficiently compute this distance.

It is also interesting to compare the d_{sym} measure with the proposals in [35]. In figure 3.2(b) the BCE and d_{sym} measures are presented for two trivial segmentations. Note the non-monotonous evolution of the BCE measure, where a monotonous behaviour (not necessarily linear) presents as the most natural. In figure 3.2(c) the evolution of the measure based in mutual information from [35] is displayed when the two segmentations being compared correspond exactly. Contrast the non-constant value of this measure, opposed to the constant value of the proposed partition distance.

3.1.2 Distance d_{sym} applied to binary partitions

What do we get if we apply d_{sym} to the edge maps? These are nothing more than binary partitions of an image in edge/non edge pixels. It is easy to prove that the value given by d_{sym} equals the number of misclassified pixels — this is the measure used in many of the earlier proposed methods.

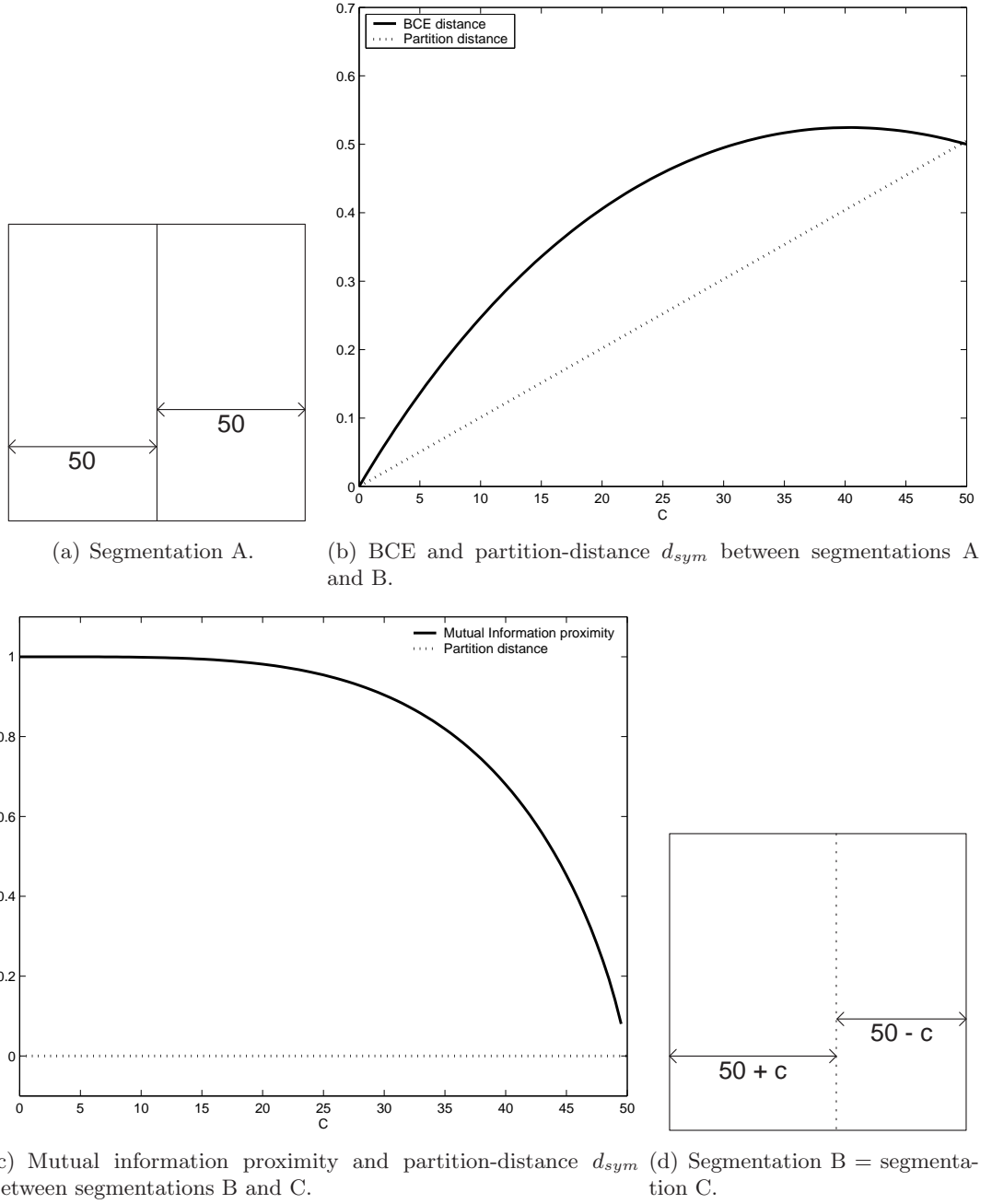


Figure 3.2: Contrast in the evolution of BCE and mutual information measures from [35] and partition-distance d_{sym} .

Proof: Let's call $C_{ne(ref)}$ and $C_{ne(eval)}$ the cluster with the non-edge pixels in the reference and under evaluation edge map, respectively; $C_{e(ref)}$ and $C_{e(eval)}$ the cluster with the edge pixels in the reference and under evaluation edge map, respectively. To equal both partitions we must either remove the points belonging to $C_{ne(ref)} \cap C_{ne(eval)}$ and $C_{e(ref)} \cap C_{e(eval)}$ or the points belonging to $C_{ne(ref)} \cap C_{e(eval)}$ and $C_{e(ref)} \cap C_{ne(eval)}$, that is,

$$d_{sym} = \min\{C_{ne(ref)} \cap C_{ne(eval)} + C_{e(ref)} \cap C_{e(eval)}; C_{ne(ref)} \cap C_{e(eval)} + C_{e(ref)} \cap C_{ne(eval)}\}.$$

Clearly, for real segmentations, the number of elements in C_{ne} , both for the reference and the under evaluation edge maps, is larger than 75% of the image's total elements. Then, their intersection must have at least 50% of the elements ($|A \cap B| = |A| + |B| - |A \cup B| \geq 75\% + 75\% - 100\% = 50\%$).

So, the minimum number of points to remove is $C_{ne(ref)} \cap C_{e(eval)} + C_{e(ref)} \cap C_{ne(eval)}$, that is, the misclassified pixels. \square

Some authors have introduced pixels distance to cope with the position of misclassified pixels in the edge map. With the proposed metric, when applied to the segmentation partition, boundaries further away from their true location imply more pixels contributing to the distance between partitions.

3.1.3 Efficient computation and graph interpretation of d_{sym}

To be of any practical use the proposed measures have to be efficient to compute. It is shown in [77] that the partition distance can be computed in polynomial time, formulating the problem as an instance of the classical assignment problem. “An instance of the classical assignment problem consists of a matrix of numbers M , and an assignment is a selection of cells of M such that no row or column contains more than one selected cell. An optimal assignment is an assignment whose selected cell values have the largest sum over all possible assignments. An optimal assignment can be computed in polynomial time as a function of the size of M . To solve the partition-distance problem, create an instance $M(P, Q)$ of the assignment problem with one row i for each cluster S_i in P and one column j for each cluster S_j in Q . Associate cell (i, j) with the subset $(S_i \cap S_j)$ and write the number $|(S_i \cap S_j)|$ in cell (i, j) . Next, solve the assignment problem on $M(P, Q)$ and let $A(P, Q)$ denote the value of the assignment. Then the partition distance equals $N - A(P, Q)$. Moreover the elements to remove from N are all those elements not associated with any selected cells of the optimal assignment.” [77]

A closer look will reveal that the above-defined matrix M is nothing more than the intersection-graph, introduced in the last chapter. Therefore, **the partition-distance is defined in the intersection-graph** as the problem of finding a maximum weight matching.

3.2 Asymmetric partition-distance, d_{asy}

In many applications under-segmentation is considered as a much more serious problem than over-segmentation. This is so because it is easier to recover true segments through a merging process after over-segmentation rather than trying to split a heterogeneous region. For those environments, it would be sensible to define an asymmetric distance between two partitions

in such a way that the distance between a partition R and any partition Q finer than R is zero. Proceeding from the theoretical foundations already built, such a measure could be tentatively defined as:

Asymmetric partition-distance, $d_{asy}(R, Q)$: given two partitions R and Q defined in a set S of N elements, the asymmetric partition-distance is the minimum number of elements that must be deleted from S , so that the induced partition Q is finer than the induced partition R . Under this asymmetric distance, any partition finer than the R partition will be at zero distance from it. Notice also that, in general, $d_{asy}(R, Q) \neq d_{asy}(Q, R)$.

Recognising that:

- a) Q is finer than R if and only if the intersection of R and Q is equal to Q
- b) $d_{sym}(Q, (R \cap Q)) = 0$ if and only if Q is finer than R

a more *ad hoc* path could be followed to define an asymmetric distance between two partitions. In fact $d_{sym}(Q, (R \cap Q))$ should, then, convey a measure of the distance from Q to a finer partition of R . But, as it is easily verified, both definitions are equivalent.

Proof that $d_{sym}(Q, (R \cap Q)) = d_{asy}(R, Q)$:

- a. Remove from S the d_{asy} elements needed to equal Q to a finer partition than R . Then in the reduced set $Q = (R \cap Q)$. That implies $d_{sym}(Q, (R \cap Q)) \leq d_{asy}(R, Q)$.
- b. Remove from S the d_{sym} elements needed to equal Q to $(R \cap Q)$ in the reduced set. Then Q is a finer partition of R in the reduced set. This implies $d_{asy}(R, Q) \leq d_{sym}(Q, (R \cap Q))$.

From a) and b) we conclude that $d_{sym}(Q, (R \cap Q)) = d_{asy}(R, Q)$. □

The maximum value this asymmetric distance can attain is also $(N - 1)$ (for instance for $Q = \text{null partition}$, $R = \text{infinite partition}$), so, to get a normalized distance we just divide by $(N - 1)$. From the definition it also follows that $d_{sym}(P, Q) \geq d_{asy}(P, Q)$.

Working with the segmentation partitions already used to exemplify the symmetric partition distance, asymmetric distance attains the values (see figure 3.3):

$$d_{asy}(\text{left}, \text{right}) = d_{sym}(\text{intersection}, \text{right}) = 10$$

$$d_{asy}(\text{right}, \text{left}) = d_{sym}(\text{intersection}, \text{left}) = 6$$

3.2.1 Efficient computation of d_{asy}

The asymmetric partition-distance, although possible to compute using the general algorithm described above and the equivalence $d_{asy}(R, Q) = d_{sym}(Q, (R \cap Q))$, can be obtained much

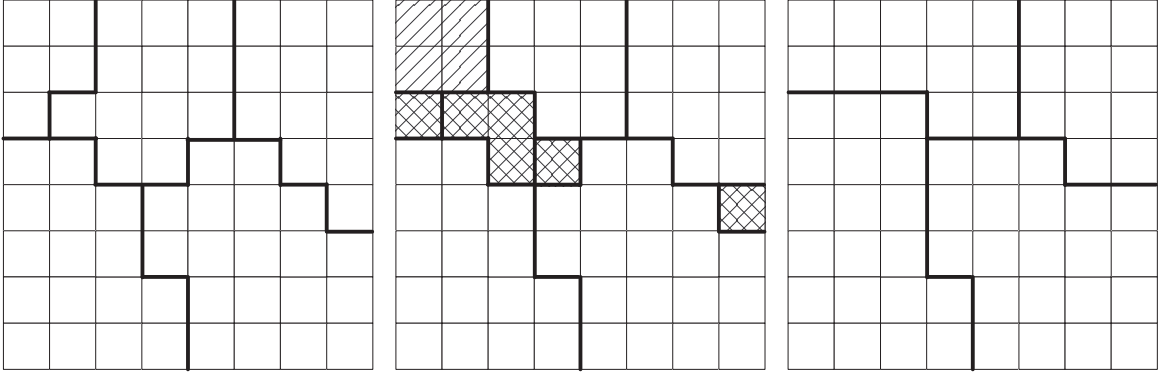


Figure 3.3: The middle partition highlights the points to be removed for the asymmetric measures.

more efficiently, realizing that $d_{asy}(P, Q) = N - \sum_i (\max_j (S_i \cap S_j))$, for all S_i in P and S_j in Q — follows directly from properties 5 and 9 of partition distance. This is readily obtained from matrix M , defined above as $N - \sum_i (\max_j M(i, j))$.

Keeping in mind that the matrix M is the intersection-graph, **the asymmetric partition-distance is also part of the general framework** proposed in the last chapter.

3.3 The mutual partition-distance, d_{mut}

In some applications, it is important to have measures tolerant to mutual refinements, [35]. It is known that humans may segment an image differently: the same scene may be distinctively perceived; different subjects may attend to different parts of the scene; subjects may segment an image at different granularities. Nevertheless, segmentations of the same image tend to be consistent in the sense that they are mutual refinements of each other [35].

A partition P is said to be a mutual refinement of a partition Q if and only if every cluster in P contains or is contained in a cluster in Q — figure 3.4.

As can easily be seen, if partition Q is a mutual refinement of partition P , then P is a mutual refinement of partition Q . This concept is easily incorporated in the proposed methodology: given two partitions P and Q defined in a set S of N elements, the mutual partition-distance, $d_{mut}(P, Q)$, is the minimum number of elements that must be deleted from S , so that the induced partitions (P and Q restricted to the remaining elements of S) are mutual refinements of each other. As easily reckoned, this is a symmetric measure.



Figure 3.4: Partitions A and B are a mutual refinement of each other.

3.3.1 Graph interpretation of the mutual partition-distance

The problem of computing the mutual partition-distance can be casted naturally as a graph problem, on the intersection-graph derived from the partitions.

We claim that partitions P and Q are a mutual refinement of each other if and only if the associated intersection-graph has only paths of length no greater than two.

Proof. Recognizing that a node (cluster) of degree one is contained in the node (cluster) to which it is connected, if we have only paths of length one or two, every node of degree greater than one is connected only to nodes of degree one — star configuration, see figure 3.5(a). Now, let $\{e_1, e_2, e_3\}$ be a path of length 3. Let v_1 be the vertex incident both to e_1 and e_2 and v_2 the vertex incident both to e_2 and e_3 . Then v_1 is not contained in v_2 (e_1 is not incident to v_2) and v_2 is not contained in v_1 (e_3 is not incident to v_1) — figure 3.5(b). \square

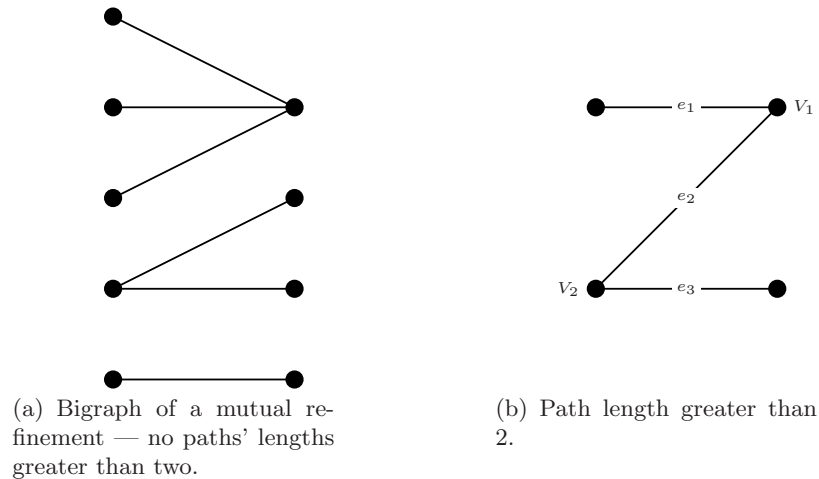


Figure 3.5: A graph corresponding to a mutual refinement has paths of length at most two.

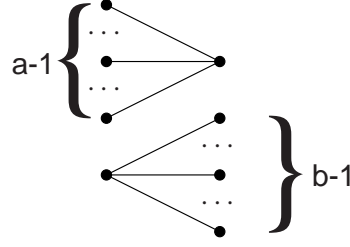


Figure 3.6: $d_{mut} \leq N - (a + b - 2)$ in $K_{a,b}$.

The mutual partition-distance can now be formulated in the associated intersection-graph as the minimum sum of weights of pruned edges that such the induced (pruned) bigraph has paths of length at most two.

The idea of modelling the relation between two segmentations of the same image by the associated bigraph has been suggested earlier by Guigues [75], where a family of nesting relations between two partitions of the same set are introduced and applied to the fusion of multi-date image segmentations. The mutual partition-distance can be seen as one element of such family of similarity measures.

3.3.1.1 Properties of the mutual partition-distance, d_{mut}

From this definition a useful set of properties can be deduced. Let P, Q, R be partitions of a set S of N elements. Then:

1. $d_{mut}(Q, P) \geq 0$ and $d_{mut}(Q, P) = d_{mut}(P, Q)$, following directly from definition.
2. The transitive property does not hold, i.e., $d_{mut}(P, Q) = 0$, $d_{mut}(Q, R) = 0 \not\Rightarrow d_{mut}(P, R) = 0$.

Consequently, the triangular inequality does not hold either.

3. Let the complete bipartite graph $K_{a,b}$, $b \geq a > 1$, $N = ab$, be the intersection-graph associated with partitions P and Q . Note that every edge has weight one. Then $d_{mut} = N - (a + b - 2)$.

Proof. That $d_{mut} \leq N - (a + b - 2)$ can be easily seen in figure 3.6, showing a possible pruning of $K_{a,b}$ leading to a graph with paths of length at most two by removing $N - (a + b - 2)$ edges.

On the other hand, a tree in $K_{a,b}$ has $a + b - 1$ edges. That implies $d_{mut} \geq N - (a + b - 1)$ — otherwise a cycle would exist in the remaining graph. Simultaneously, any subgraph

$K_{2,2}$ of $K_{a,b}$ cannot be connected after pruning, as is trivial to verify. So $d_{mut} \geq N - (a + b - 2)$. \square

4. Let $\mathcal{BG}_{a,b}$, $a = \lceil \frac{N}{\sqrt{N}} \rceil$, $b = \lceil \sqrt{N} \rceil$, be the intersection-graph associated with partition P and Q , with every edge with weight one. Then $d_{mut} = N - (a + b - 2)$.

Proof. • $d_{mut} \geq N - (a + b - 2)$ — as in last item.

- $d_{mut} \leq N - (a + b - 2)$
 - $N > a(b - 1)$ so at least 1 node from P is connected to all nodes from Q .
 - $N > (a - 1)b$ so at least 1 node from Q is connected to all nodes from P .

Then, as in last item, we can keep $(a - 1) + (b - 1)$ edges.

\square

5. From last item, given a set of N elements, we can always find two partitions $N - (\lceil \frac{N}{\sqrt{N}} \rceil + \lceil \sqrt{N} \rceil - 2)$ elements apart. So, $d_{mut}^{max} \geq N - (\lceil \frac{N}{\sqrt{N}} \rceil + \lceil \sqrt{N} \rceil - 2)$. Normalizing the mutual partition-distance by the same factor as the partition-distance was normalized, $N - 1$, gives a normalized distance, ranging from 0 to approximately 1, for typical values of N in image segmentation (N equals the number of pixels in the image): $\frac{N - (\lceil \frac{N}{\sqrt{N}} \rceil + \lceil \sqrt{N} \rceil - 2)}{N - 1} \approx 1$.

3.3.2 Connection to the partition-distance

Besides the mutual partition-distance, a set of different measures to evaluate the quality of an image segmentation Q , when comparing it to a reference segmentation R , have already been proposed:

- a symmetric measure $d_{sym}(R, Q)$, the partition-distance between the two partitions, given by the minimum number of elements that must be deleted from the original set S , so that the two induced partitions in the remaining elements are identical.
- an asymmetric measure $d_{asy}(R, Q) = d_{sym}(R \cap Q, Q)$, tolerant to over-segmentations, given by the minimum number of elements that must be deleted from S , so that the induced partition Q is finer than the induced partition R .
- an asymmetric measure $d_{asy}(Q, R) = d_{sym}(R \cap Q, R)$, tolerant to under-segmentations, defined similarly to $d_{asy}(R, Q)$.

As already established, the computation of d_{sym} is mapped in the traditional matching problem, in the corresponding intersection-graph; $d_{asy}(R, Q)$, a special and rather straightforward

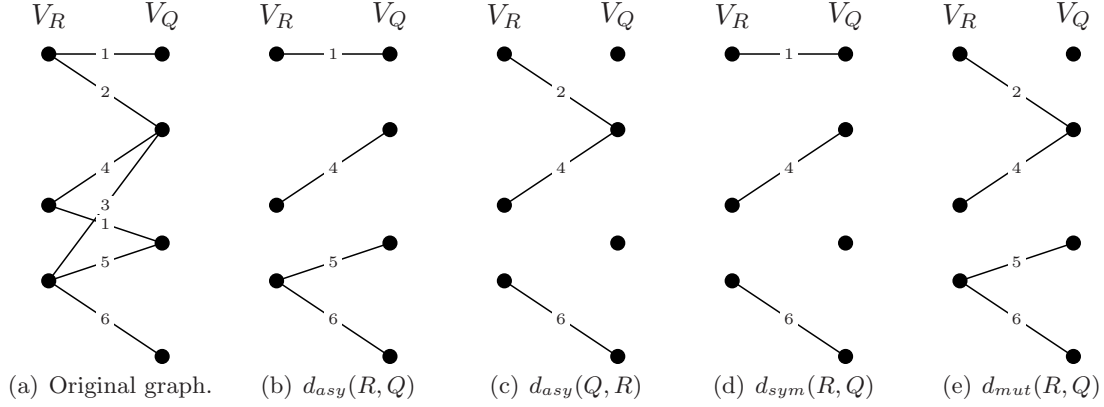


Figure 3.7: Original and resulting graphs for all measures.

instance of the matching problem, is simply translated as “for each vertex in V_Q remove all but the edge of biggest weight”; $d_{asy}(Q, R)$ is computed in a similar way — figure 3.7.

The resulting graph for $d_{asy}(R, Q)$ is partitioned in disconnected $K_{1,m}$ subgraphs with the star center always in V_R . The resulting graph for $d_{asy}(Q, R)$ has the star centers in V_Q . The original partition-distance does not allow stars (or only degenerate stars $K_{1,1}$). The mutual partition-distance weakens the constraints on the number and position of the star centers.

It is not hard to prove that, by keeping constant the number and position of the star centers (possibly some in V_R and others in V_Q), the problem of computing the mutual partition-distance simplifies to a matching problem.

Proof. Start by labelling arbitrarily each vertex either as a star center or as a leaf. Impose the following additional constraints:

- two vertices labelled as star centers can not be connected by an edge
- a vertex labelled as a leaf can not have degree greater than one

The mutual partition-distance, constrained as above, simplifies to a matching problem (figure 3.8):

1. remove every edge connecting two star centers;
2. for each leaf, remove every edge to a star center, except the biggest of them (the others are *dominated* by this) — this step is not strictly necessary;
3. split every star center S_i in $\deg(S_i)$ vertices, each with one and only one of the $\deg(S_i)$ edges;

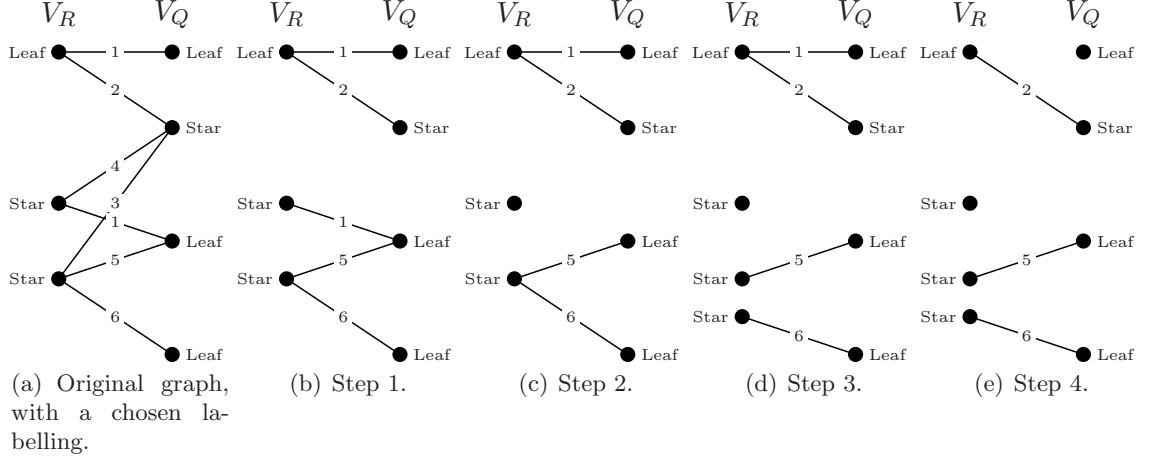


Figure 3.8: Mutual partition-distance with constrained star centers.

4. perform the traditional matching in the resulting graph.

Because the result of such algorithm is in fact a mutual refinement in the original graph, then d_{mut} is less or equal than the result of any labelling.

It is also easily reckoned that the mutual partition-distance equals one of such labelling: in the resulting graph for d_{mut} , label each vertex with degree greater than one as star center and the others as leaves. Then d_{mut} is equal to the result for this particular labelling. \square

We see that the mutual partition-distance can be interpreted as a generalization of the partition-distance problem.

It should also be apparent that the number of stars in the optimal solution will be no greater than $\min(\#V_R; \#V_Q)$: every star makes use of, at least, one vertex of V_R and V_Q .

3.3.3 Mutual partition-distance as an optimization problem

The computation of the mutual partition-distance can be performed directly on the corresponding bipartite graph by removing every possible combination of clusters' intersections (edges) and assessing the validity of the resulting graph. The search space can be traversed using a gray code counter [78]: in each iteration only a single edge needs to be added or removed to the graph under evaluation. This leads to an exponential-time algorithm, as a function of the number of clusters' intersections (number of edges of the graph). Another possibility would be to use the results of section 3.3.2, and compute every possible matching problem. This would lead to an exponential-time algorithm, as a function of the number of clusters (number of vertices of the graph). However, it is easy to show that the computation of the mutual partition-distance fits the definition of an integer optimization problem.

For the mathematical model, use the following decision variables:

$\mathbf{X} = \mathbf{1} - \mathbf{Y}$, with $\mathbf{X} = [x_1, \dots, x_n]^T$, $\mathbf{Y} = [y_1, \dots, y_n]^T$, and $x_i = \begin{cases} 1 & \text{if edge } e_i \text{ is kept} \\ 0 & \text{if not} \end{cases}$,
for each edge e_i .

Setting $\mathbf{W} = [w_1, \dots, w_n]^T$, where w_i is the weight of edge e_i , we formulate the mutual partition-distance as the following integer constrained minimization problem:

$$\begin{aligned} d_{mut} &= \min \mathbf{W}^T \mathbf{Y} \\ \text{s.t. } & y_i + y_j + y_k \geq 1, \text{ for each trio of edges} \\ & e_i, e_j, e_k \text{ forming a path of length 3} \\ & y_i \in \{0, 1\} \end{aligned} \tag{3.1}$$

For $K_{a,b}$ this translates in $a \cdot b$ decision variables and $a(a-1)b(b-1)$ constraints. However, noting that a graph corresponds to a mutual refinement if and only if every $\mathcal{BG}_{2,2}$ subgraph has, at most, two edges, the mutual partition-distance can be alternatively defined as:

$$\begin{aligned} d_{mut} &= \min N - \mathbf{W}^T \mathbf{X} \\ \text{s.t. } & \sum_{e_i \in \mathcal{BG}_{2,2}} x_i \leq 2, \text{ for each } \mathcal{BG}_{2,2} \text{ subgraph} \\ & x_i \in \{0, 1\} \end{aligned} \tag{3.2}$$

This reduces the number of constraints in $K_{a,b}$ to $\frac{a(a-1)b(b-1)}{4}$. This formulation has the additional benefit of reducing the set of continuous feasible solutions. In fact, although both formulations are equivalent for binary variables, they would differ if the decision variables were relaxed to $[0,1]$. As easily reckon, the feasible solution set for the second formulation would be a subset of the first. This may result in a faster convergence of the second formulation, as algorithms for solving the integer problem are usually based on the continuous counterpart formulation.

3.3.3.1 Reformulation with a compact convex domain

Formulations (3.1) and (3.2) are a brute force NP-hard integer minimization problem. In general, there is no efficient way of (optimally) solving such type of problems. Nonetheless,

it can be shown to be equivalent to the following concave minimization problem [79]:

$$\begin{aligned}
 d_{mut} &= \min \mathbf{W}^T \mathbf{Y} + \mu \mathbf{Y}^T (\mathbf{1} - \mathbf{Y}) \\
 \text{s.t.} \quad &y_i + y_j + y_k \geq 1, \text{ for each trio of edges} \\
 &\quad \quad \quad e_i, e_j, e_k \text{ forming a path of length 3} \\
 &y_i \in [0, 1]
 \end{aligned} \tag{3.3}$$

where μ is a sufficiently large positive number. Provided that μ is large enough the global minimum is attained only when $\mathbf{Y}^T (\mathbf{1} - \mathbf{Y}) = 0$.

3.3.3.2 Reformulation as a generalization of the partition-distance

Yet another formulation can be devised, by adopting a different viewpoint, based on the results of section 3.3.2. Introduce the additional binary variables

$$\begin{aligned}
 r_i &= \begin{cases} 1 & \text{if vertex } vr_i \in V_R \text{ is a star center,} \\ 0 & \text{if not} \end{cases}, \text{ for each vertex } vr_i \in V_R. \\
 c_j &= \begin{cases} 1 & \text{if vertex } vc_j \in V_C \text{ is a star center,} \\ 0 & \text{if not} \end{cases}, \text{ for each vertex } vc_j \in V_C.
 \end{aligned}$$

The mutual partition-distance can now be computed as

$$\begin{aligned}
 d_{mut} &= \min N - \mathbf{W}^T \mathbf{X} \\
 \text{s.t.} \quad &x_i + r_{e_i} + c_{e_i} \leq 2, \forall \text{ edge } e_i, \text{ with} \\
 &\quad \quad \quad vr_{e_i} \text{ and } vc_{e_i} \text{ incident to } e_i \\
 &\sum_{\forall e_i \text{ incident to } vr_k} x_i \leq 1 + r_k \cdot \deg(vr_k) \\
 &\sum_{\forall e_i \text{ incident to } vc_k} x_i \leq 1 + c_k \cdot \deg(vc_k) \\
 &x_i, r_i, c_j \in \{0, 1\}
 \end{aligned} \tag{3.4}$$

The first condition expresses that two star centers can not be connected; the second and the third that if a vertex is a leaf, it can only have a edge incident with it — if the vertex is a star center the inequality is always satisfied.

This formulation requires $ab + a + b$ decision variables and the same number of constraints in $K_{a,b}$. In this formulation the variables x_i can be relaxed to the real domain $[0,1]$. This follows directly from the established correspondence with the matching problem which, as well known, can be solved in the continuous domain $[0,1]$.

Because the efficiency of each formulation depends on the application, one should select the most suitable for the target task.

3.4 Proposed discrepancy measures

The path covered so far leads us to propose a set of different measures to evaluate the quality of an image segmentation S when comparing it to a reference segmentation R :

- Generic discrepancy measure given by the normalized partition-distance between the reference segmentation and the segmentation under study: $d_{sym}(R, S)/(N - 1)$, where N is the number of pixels in the image.
- Asymmetric-measure for applications where over-segmentation is not an issue, $d_{asy}(R, S)/(N - 1)$, where R is the reference segmentation, S is the segmentation to assess and N is the number of pixels in the image.
- Asymmetric-measure for applications where under-segmentation is not an issue, $d_{asy}(S, R)/(N - 1)$, where R is the reference segmentation, S is the segmentation to assess and N is the number of pixels in the image.
- Mutual partition-distance, $d_{mut}(S, R)$, when mutual refinements can be tolerated.

It should not be difficult to further extend this framework according to the specificities of each application.

3.5 Experiments

The proposed metrics were applied to a set of segmentations outputted by some selected segmentation algorithms and results were compared in order to assess the metrics' quality. For that end, a software application[†] was developed to implement the proposed metrics. The assignment problem was solved based on the well-known hungarian method by Kuhn [80]. For HD images (1920×1080) with less than 256 regions, the computation takes less than one second in a regular PC (1 GHz AMD microprocessor, 256 MB RAM).

In a first test to check the adequacy and performance of the proposed solution to evaluate segmentation's quality, the symmetric metric was applied to the output of two range segmentation algorithms (UE and USF) presented in [68], using the ABW imagery, provided by the same author. The distance from the ground truth segmentation and the segmentation produced by each algorithm was calculated for each of the 30 test images on the set.

The partition distance results, presented in figure 3.9, consistently attribute better quality to UE, except for the 15th frame. This rating was found consistent with the subjective

[†]The software, as well as all streams used in the tests, is available upon request to the authors.

evaluation that a human observer would make by direct visualization of the segmentation partitions. These results are also in accordance with the average values in [68].

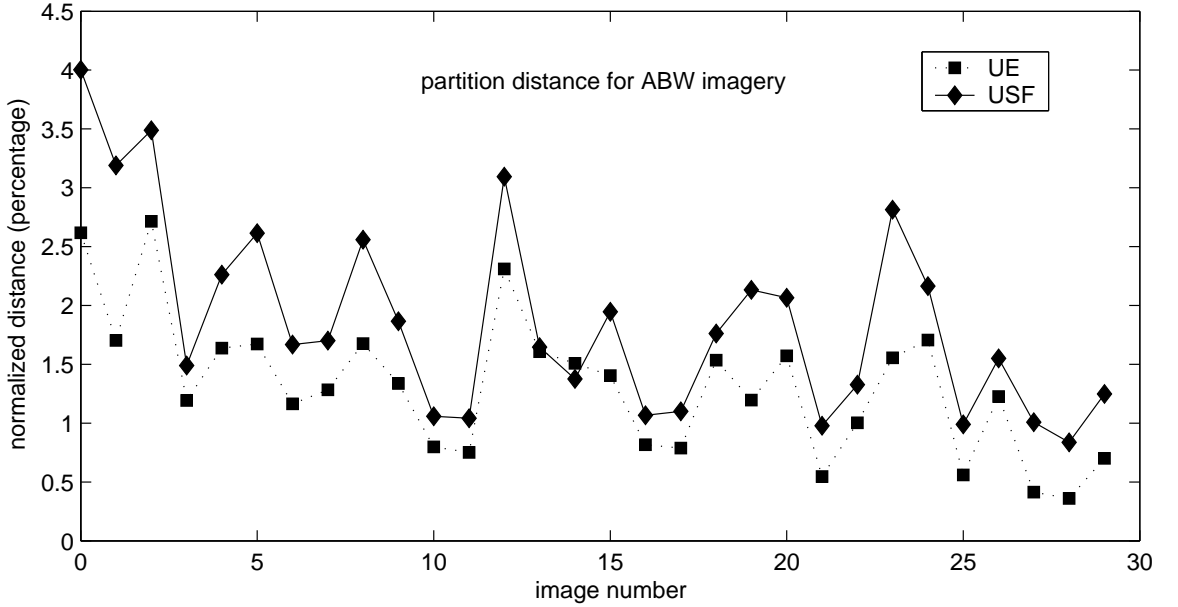


Figure 3.9: UE versus USF results.

In a second test, the strength of the proposed asymmetric distances was also gauged. Towards this end, a segmentation algorithm that can be parametrically configured was selected. Different segmentation partitions, S_n , were produced for the same image (see figure 3.10), where n stands for the number of regions obtained for the partition. For each pair of segmentation partitions we computed the d_{sym} and d_{asy} distances. Results are presented in table 3.5.



Figure 3.10: Segmentation partition S_{23} on left, segmentation partition S_{226} on right, original image on center.

From table 3.5 we see that d_{sym} increases as we move away from the main diagonal. This is expected because as $|i - j|$ increases S_i and S_j become more and more different. However, for a given S_i , $d_{asy}(S_i, S_j)$ decreases while j increases until i , attaining 0 when $j = i$. It then stabilises in very low values for $j > i$. This is so because segmentation algorithms tend to produce finer partitions as the segmentation resolution is increased.

Table 3.1: Tables showing the symmetric and asymmetric results (percentage values).

(a)

d_{sym}	S_{23}	S_{47}	S_{77}	S_{84}	S_{129}	S_{226}
S_{23}	0.00	14.66	30.09	30.10	34.95	51.30
S_{47}	14.66	0.00	18.84	18.86	23.92	40.73
S_{77}	30.90	18.84	0.00	0.09	12.89	30.87
S_{84}	30.10	18.86	0.09	0.00	12.88	30.85
S_{129}	34.95	23.92	12.89	12.88	0.00	25.12
S_{226}	51.30	40.73	30.87	30.85	25.12	0.00

(b)

d_{asy}	S_{23}	S_{47}	S_{77}	S_{84}	S_{129}	S_{226}
S_{23}	0.00	0.85	1.80	1.80	2.10	2.20
S_{47}	14.61	0.00	2.98	2.98	3.03	3.25
S_{77}	29.98	18.21	0.00	0.00	4.20	3.67
S_{84}	29.99	18.23	0.09	0.00	4.24	3.70
S_{129}	34.68	23.39	10.51	10.46	0.00	3.76
S_{226}	51.11	40.32	30.16	30.11	24.74	0.00

Finally, the mutual partition distance was assessed with the Berkeley Segmentation Dataset [36]. The dataset consists of a collection of images where each image was segmented by different humans in color, grayscale and inverted-negated [35].

We implemented both d_{mut} and Martin's LCE measure [35] in C++. Formulation (3.2) of the mutual partition-distance was used in the software implementation. The linear programming problem was solved with the freely available *lp_solve vs 5.0* software. Tests were carried out on a regular PC (1 GHz AMD microprocessor, 256 MB RAM).

Figure 3.11 shows the distribution of d_{mut} and d_{sym} over the dataset for pairs of segmentations of the same image and pairs of segmentations of different images. As seen, although two segmentations of the same image may differ appreciably, as given by the d_{sym} measure, they are almost always identical, in what concerns the d_{mut} measure. For segmentations of the same image, the mutual partition-distance exhibits a strong peak near zero error, given evidence of the consistency of human segmentations. It is also visible that the fraction of overlap — bayes risk — is smaller for the mutual partition-distance. Some examples of segmentation pairs, at different values of d_{sym} and d_{mut} , are shown in figure 3.12, each distance being presented both as a numerical value and as black pixels of a mask image. These results are in accordance with the results achieved in [35].

Then, we proceeded with the comparison of d_{mut} and LCE [35], in the capability of discriminating same-image and different-image pairs of segmentations.

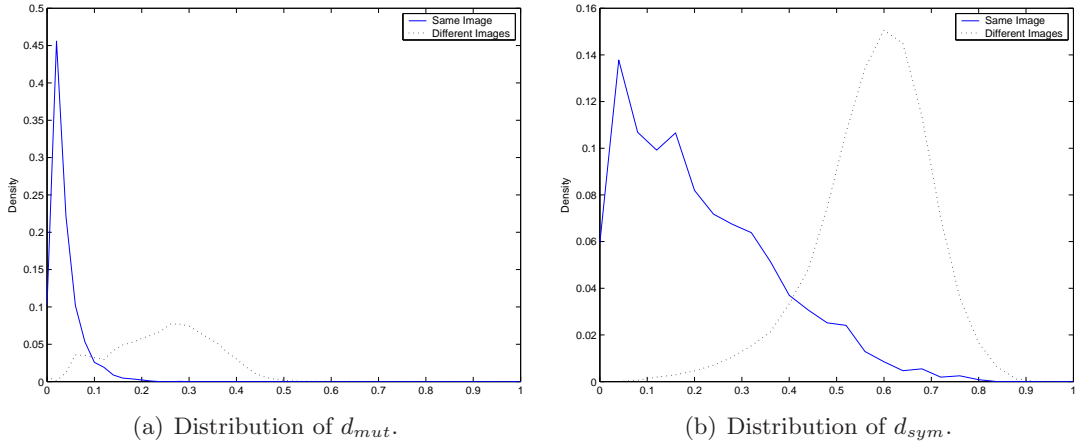


Figure 3.11: Comparison of d_{mut} and d_{sym} for pairs of human segmentations.

In figure 3.13(a) we plot the distribution of the d_{mut} and the LCE measures over the segmentation database, for pairs of segmentations of the same image; in figure 3.13(b) it is presented d_{mut} vs. LCE for pairs of segmentations of the same image. In figure 3.14 the same information is depicted for pairs of segmentations of different images. As expected, both measures are portraying similar information.

It may seem a bit disappointing, however, that the proposed measure has an inferior capability to discriminate segmentations of the same image from segmentations of different images, than the LCE measure, as evaluated by the Bayes Risk, figure 3.15. That is probably a consequence of some human segmentation inconsistencies or errors (second pair in figure 3.16) and of degenerate pairs in the different-image pairs: segmentations that compare favourably with nearly any other because all the segments are small or fortuitous alignment of segmentations [35].

Although the proposed measure was not intended for the separation of same-image and different-image pairs of segmentations, it is not difficult to, using the general setting introduced in section 2.3.1, construct a measure with improved performance for this task.

To exemplify, and starting from the intersection-graph corresponding to a pair of segmentations, we considered as an additional feature the percentage of remaining edges in the calculation of d_{mut} , d_{rem} . Following a pattern classification approach [81], based on the SVM principle [82], several pairs of features were gauged: (d_{mut}, LCE) , (d_{mut}, d_{rem}) , (LCE, d_{rem}) . An optimized measure for the separation of these two populations was obtained as $d_{opt} = 0.82d_{mut} + 0.18d_{rem}$, with the boundary decision at $d_{opt} = 0.165$, figure 3.17. Naturally, better measures for this dataset could be thought, either by adopting non-linear boundaries in the selected two-feature space or by considering other features.

A side information of the proposed measures is the indication of the erroneous pixels (the



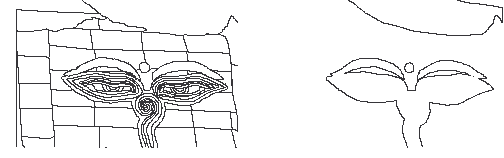


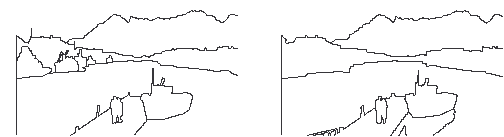


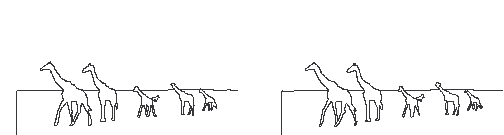
d_{sym}	d_{mut}	pair of segmentations
0.8002 	0.0131 	
0.0825 	0.0217 	
0.0085 	0.0082 	

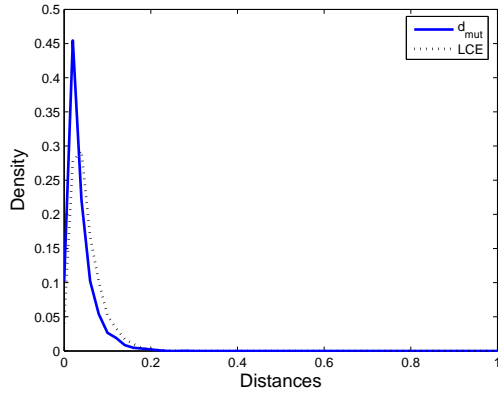
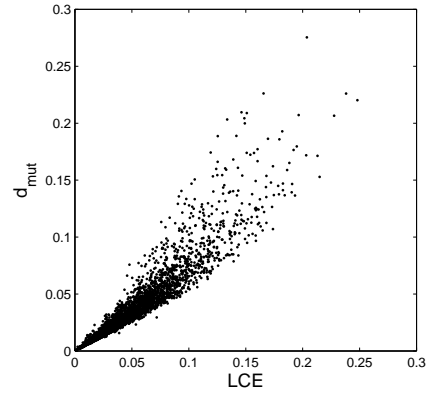
Figure 3.12: Example pairs at various d_{sym} and d_{mut} values.(a) Distribution of the d_{mut} and LCE [35] measures.(b) d_{mut} vs. LCE [35].

Figure 3.13: Comparison over the Berkeley Segmentation Dataset for pairs of human segmentations of the same image.

pixels to be removed), black pixels of the mask images in figures 3.12 and 3.16, information that can be useful for further processing.

3.6 Discussion

In this chapter we have instantiated several measures to compare image segmentations based on the intersection-graph. Starting from the symmetric partition-distance and the asymmetric partition-distance, for both of which efficient computing algorithms exist, we

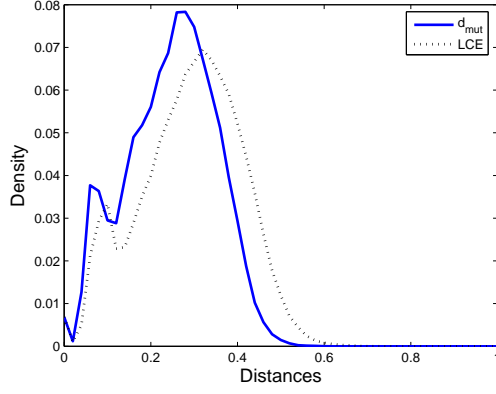
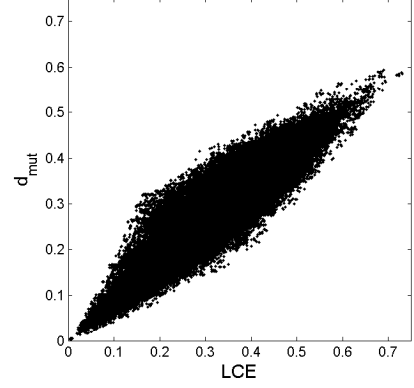
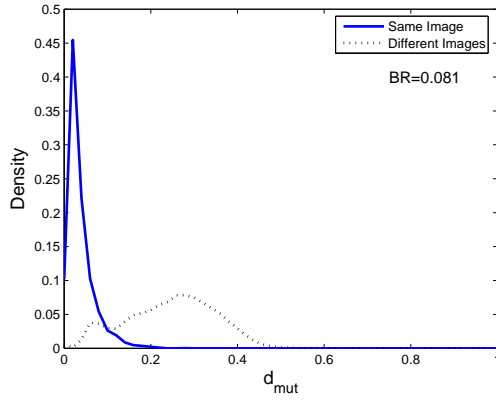
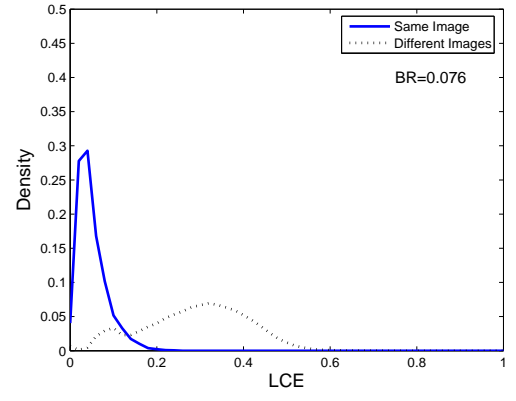
(a) Distribution of the d_{mut} and LCE [35] measures.(b) d_{mut} vs. LCE [35].

Figure 3.14: Comparison over the Berkeley Segmentation Dataset for pairs of human segmentations of different images.

(a) Bayes Risk for the d_{mut} measure.

(b) Bayes Risk for the LCE measure.

Figure 3.15: Comparison of the Bayes Risk.

concluded with the mutual partition-distance. A link has been established between this measure and the partition-distance, which can be seen as a special case of the mutual partition-distance. Binary integer linear programming formulations for the computation of the measure were also provided. The resulting algorithms have shown to be effective when applied to a real-world dataset. Although the notion of mutual refinement may not capture all the unpredictability in human segmentations, it certainly models a wide category of variabilities. This may imply the need to complement it with other criteria, as exemplified by the two feature example.

The aim of this work is not to propose an evaluation measure incorporating perceptual or contextual information. As a low level measure, the proposed techniques should rather





LCE	d_{mut}	pair of segmentations	
0.13	0.08		
0.13	0.20		

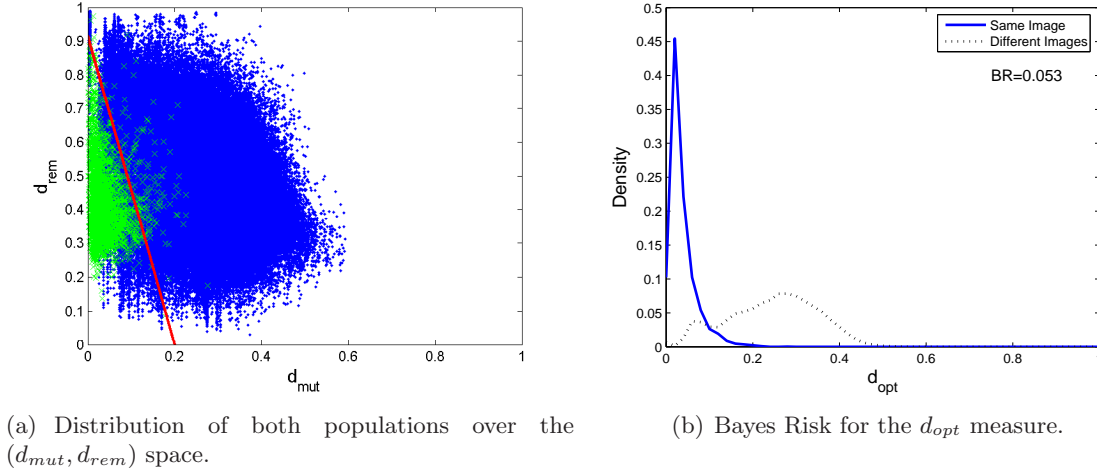
 Figure 3.16: Example pairs at various LCE and d_{mut} values.


Figure 3.17: Construction of a measure with improved discriminative capability.

produce valid results under all applications where the reference segmentation is available. These measures could also be used as a building block in more complex and application specific evaluation schemes.

Chapter 4

Data in, data out fusion approaches for hybrid image segmentation

Multisensor data fusion has been traditionally characterized as integration at different hierarchical levels depending on the stage of the processing at which such data fusion takes place. It has been common practice to view this as a three-level hierarchy, namely, data fusion, feature fusion, and decision fusion. Under this umbrella, general and well-known techniques for data fusion could be attempted for hybrid image segmentation.

In this chapter, after providing an overview of the different schemes for fusion of information, we evaluate fusion techniques based on the data in, data out fusion model. This model is the most straightforward approach, as it allows the use of standard and established segmentation algorithms without modification, by creating a new colour triplet containing information from the original colour and depth images. The general approach for the fusion of images based on the data in, data out fusion model, is to create a new set of images, usually of reduced number, from the original set of images, figure 4. All images are assumed to be geometrically aligned and have the same pixel size.

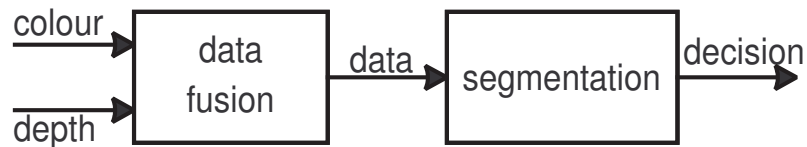


Figure 4.1: Data in, data out fusion.

4.1 Multisensor information fusion[§]

In a fusion process, information may be of various kinds, ranging from measurements to verbal reports. Some data cannot be quantified; their accuracy and reliability may be difficult to assess. Accordingly, the definition of data fusion should not be restricted to data output from sensors (signal). Opposite to most of the published definitions, it should not be restricted to methods and techniques or refer to functional models or architectures of systems.

Under the auspices of the SEE, the French affiliate of the IEEE, and the EARSeL, the European affiliate of the International Society for Photogrammetry and Remote Sensing (ISPRS), the following definition was agreed in 1998:

data fusion is a formal framework in which are expressed the means and tools for the alliance of data originating from different sources. It aims at obtaining information of greater quality; the exact definition of ‘greater quality’ will depend upon de application.

Note that the word “data” in data fusion is taken in a broad sense; It may be replaced by information fusion.

If observations are provided by sensors and only by sensors, one use the term *sensor fusion* or *multisensor fusion*. Image fusion is a sub-class of sensor fusion; here the observations are images. This will be the sub-domain of interest from now on.

In general, one can look upon sensors as windows into the physical environment in which the phenomenon under observation is occurring over a period of time, with each sensor having its own uniquely defined window. These windows, in effect, are constraints on what they can sense or perceive, or measure. Accordingly, the information generated in the environment, about the phenomenon under observation by the sensors can be thought of as undergoing decomposition into its components by the sensors; that is, sensor (caused) fission. This information fragmentation, resulting from such an unavoidable fission process, has to be appropriately counteracted by a sensor (information) fusion process. This supports the postulate that fusion is a fission inversion process and forms the basis for development and assessment of sensor fusion methodologies. Stated equivalently, these individual information components, acquired through the different sensor windows, require a reunification, that is, sensor fusion, to derive the factual representation of the phenomenon occurring in the environment. Here, fission is a natural occurrence resulting from the deployment of real-world sensors with specific physical constraints, both spectral (in terms of what they can see) as well as spatial (in terms of where and how far or near they can observe). Thus fission,

[§]Compiled from [83–87].

being unavoidable, has to be counteracted with a suitably "optimal" fusion process. The search for the optimal fusion process should therefore be as broad as practical and has to explore all potentially beneficial avenues spanning all the conceivable fusion models. This represents the main objective of the ensuing discussions. Thus this sensor fusion activity can be looked upon as an information retrieval or preservation process and one can conceive of an associated measure of effectiveness. An ideal sensor fusion process would therefore be able to retrieve or restore all of the inherent information of interest in the environment from the data sensed by the multisensor suite.

Registration

The information entering a fusion process should be aligned. The alignment of sources defines a common representation on the basis of the measurements and the representations at an instant of time. Many techniques for image fusion are available [88, 89]. However, we will assume that images have already been delivered co-registered.

4.1.1 Fusion classification

Sensor fusion concepts and techniques can be characterized from a variety of perspectives, such as application domain (intended application, such as defence, robotics, medical, and space), fusion objective (detection of an object or an event occurrence, recognition of an object class or event category, identification of an object or characterization of an event, tracking of an object or monitoring an event, estimation of a future state of a system, achieving physical contact with an object, taking note of phenomenological changes, assessment of quality/quantity in processes, and combining data sources to make decisions), sensor type (active sensors, passive sensors, human sources, data archives, etc), sensor suite configuration (parallel and serial or tandem configurations are by far the most common), and fusion level.

Multisensor data integration has been traditionally characterized as fusion at different hierarchical levels depending on the stage of the processing at which such data integration takes place. It has been common practice to view this as a three-level hierarchy, namely, data fusion, feature fusion, and decision fusion. This three level hierarchy has become fairly accepted terminology although to some extent this is still a matter of individual choice and hence is subjective in nature. However, in the aforementioned fusion levels, the input and output are assumed to be at the same level. But that does not need to be the case. Expanding the three level hierarchy of fusion into five fusion process I/O dependent modes, we have [83]:

1. Data in, data out (DAI-DAO) fusion. This is the most elementary or lowest form of fusion conceivable under this hierarchy. This fusion mode, being one of fusion of data inputs resulting in a form of data output, has been commonly referred to as data fusion. This could conceivably be used in combining information from like sensors, that is, sensors with compatible data rates, data dimensionality and formats, for other application areas as well. Of course data registration, both spatial and temporal, is critical to successful data fusion. Fusion paradigms in this category are generally based on techniques developed in the traditional signal and image processing domains, such as arithmetic (linear or non-linear) or logical operations that combine two or more similar data elements to produce another similar fused data element.
2. Data in, feature out (DAI-FEO) fusion. Here, data from different sensors (or different bands of the same sensor) are combined to derive some form of a feature of the object in the environment or a descriptor of the phenomenon under observation. Fusion in this mode, depending on one's view point, input-fusion of data or output-fusion resulting in features, has been looked upon either as data fusion or feature fusion. The manner in which depth perception is achieved in humans, by combining the visual information acquired from the two eyes, can be looked upon as a classical paradigm of this feature or information fusion.
3. Feature in, feature out (FEI-FEO) fusion. Both the input and output of the fusion process are features. Accordingly, this has been commonly referred to as feature fusion. Typically under feature or discriminant fusion, instead of sensed measurements, derived features are combined either quantitatively, say, in a multidimensional feature space sense or qualitatively, within a heuristic decision logic process, or through a combination of such qualitative and quantitative information. This is particularly true when each sensor in the environment has its own set of uniquely different data structure and features obtainable from one are not derivable from the other. For example, shape features obtainable from an imaging sensor may not be available from a nonimaging radar and on the other hand, range information obtainable from the latter may be outside the scope of the former. The two pieces of information can be combined to derive a measure of the volumetric size of a target, a typical fusion activity at the feature level.
4. Feature in, decision out (FEI-DEO) fusion. Here, the inputs are the features from different sensors and the output of the fusion process is a decision such as target class recognition. Here, the fusion process accepts features extracted from different sensors and derives decisions based on a simultaneous assessment of the multisensor based features. Most of the tools available for processing in this mode are based on classical pattern recognition concepts and can be looked upon as merely the natural extension to the domain of multi-source data from its classical use in the context of single source data.

5. Decision in, decision out (DEI-DEO) fusion. In this mode the fusion process essentially combines decisions made from independent data sources and/or independent decision processes operating on different aspects of the data from the same source to generate a more robust decision that is less sensitive to the vagaries of the individual decision processes. A variety of approaches exist in this domain and include voting schemes (majority based fusion - single look and multi-look temporal fusion, consensus based fusion — multilook temporal fusion, combinations of Boolean logic, simple and weighted voting), probabilistic approaches (Bayesian, Neyman-Pearson, etc.), and other miscellaneous approximate reasoning approaches such as, fuzzy logic, evidential reasoning, neural nets. The applications discussed below show the extent to which these predominate in recent studies. In real-world applications, fusion potential may exist in more than one of these modes and has to be exhaustively explored.

There exists a persistent school of thought among researchers in the sensor fusion arena that fusion at the lowest possible level in a given scenario is the best approach since the level of detail in the information is highest at that level. However, it should be noted that the corruption of information due to noise is also the highest at that level. The process of extracting relevant information from the data, in terms of features and decisions, on the one hand may be throwing away valuable information in terms of the details, but on the other may also be reducing the noise that degrade the quality of the ensuing decision. Thus there is an obvious trade-off to be evaluated in choosing the fusion architecture best suited for a given application scenario. Only an exhaustive exploration of the alternatives individually and in combination can help in evaluating such trade-off.

4.2 Intensity substitution approach for hybrid image segmentation

The intensity substitution approach is based on substituting the intensity (I) from the colour information by a linear combination of the depth image (D) and the original intensity: $I' = \alpha D + (1 - \alpha)I$, $\alpha \in [0, 1]$. Toward that end the colour information is previously converted into a convenient colour space, such as IHS or $L^*u^*v^*$, where the brightness is decoupled from the hue and colour purity. The substitution of the intensity implies to previously match the dynamic range of D to I , which can be done by histogram matching [90], or variance and mean matching, or other techniques. Finally, $(I', *, *)$ is converted back into the original colour model.

4.2.1 Results

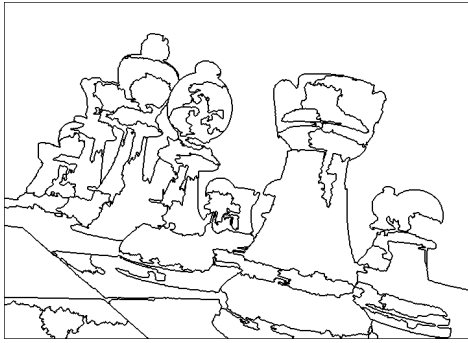
Figures 4.2, 4.3 and 4.4 exhibit some selected results obtained with the intensity substitution method. RGB values were converted to YC_bC_r and the dynamic range of depth data was matched to Y with the algorithm proposed in [90].



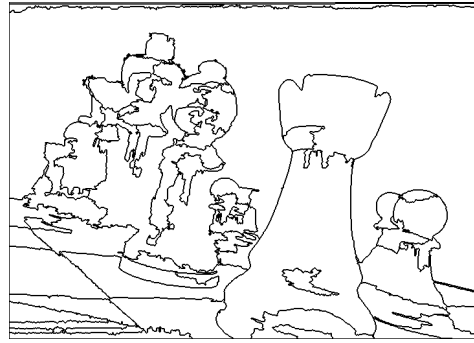
(a) Fused 'chess' image, $\alpha = 0.5$.



(b) $\alpha = 0.0$. $d_{sym} = 48.77$, $d_{mut} = 10.40$



(c) $\alpha = 0.5$. $d_{sym} = 47.32$, $d_{mut} = 7.73$



(d) $\alpha = 1.0$. $d_{sym} = 45.21$, $d_{mut} = 11.75$

Figure 4.2: Results with the Mean Shift algorithm, for the 'chess' image.



(a) fused 'billiards' image, $\alpha = 0.5$.



(b) $\alpha = 0.0$. $d_{sym} = 62.73$, $d_{mut} = 3.32$



(c) $\alpha = 0.5$. $d_{sym} = 59.23$, $d_{mut} = 6.38$

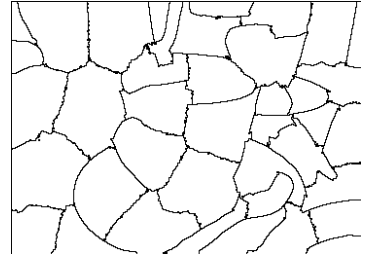


(d) $\alpha = 1.0$. $d_{sym} = 58.08$, $d_{mut} = 7.09$

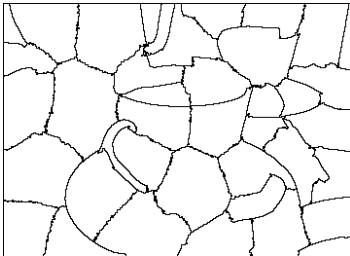
Figure 4.3: Results with the JSEG algorithm, for the 'billiards' image.



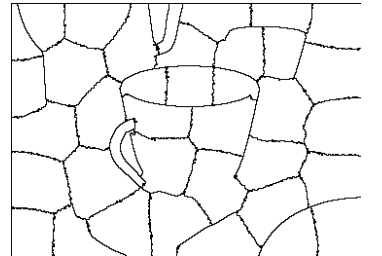
(a) Fused 'teacup' image, $\alpha = 0.5$.



(b) $\alpha = 0.0$. $d_{sym} = 73.04$, $d_{mut} = 9.70$



(c) $\alpha = 0.5$. $d_{sym} = 73.17$, $d_{mut} = 10.35$



(d) $\alpha = 1.0$. $d_{sym} = 78.54$, $d_{mut} = 11.21$

Figure 4.4: Results with the NCut algorithm, for the 'teacup' image.

4.3 Multiresolution approach for hybrid image segmentation

As seen, the most straightforward approach to image fusion is to take a weighted average of the intensity and depth images pixel by pixel; however, along with simplicity comes several undesired effects including a loss of detail.

It was recognized that multiscale transforms (MST) could be very useful for analyzing the information content of images for the purpose of fusion. Most of these approaches are based on combining the multiscale decompositions (MSD's) of the source images, figure 4.5. The basic idea is to perform a MSD of each source image, by applying a MST. This transformation

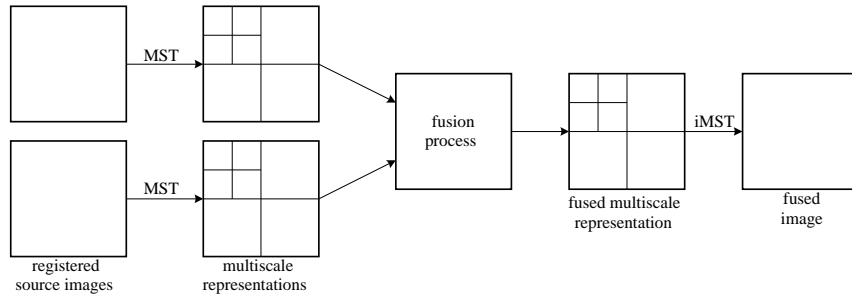


Figure 4.5: Block diagram of a generic image fusion scheme (from [91]).

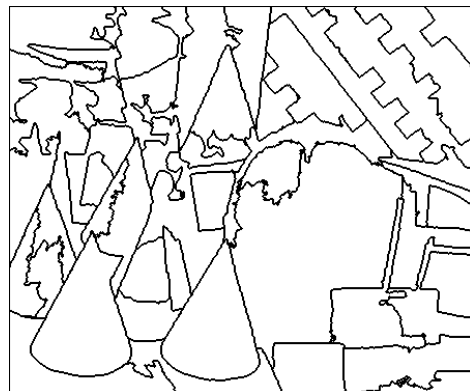
domain representation emphasizes important details of the source images at different scales, which is useful for choosing the best fusing rules. Then, using a feature selection rule, a fused multiscale representation is formed from the pair of multiscale representations. The simplest feature selection rule is choosing the maximum of the two corresponding transform values. This allows the integration of details into one image from two or more images. Finally a fused image is obtained by taking the inverse multiscale transform (iMST) of the fused representation. Pyramid transform and discrete wavelet transform are the most commonly used MSD methods [91].

4.3.1 Results

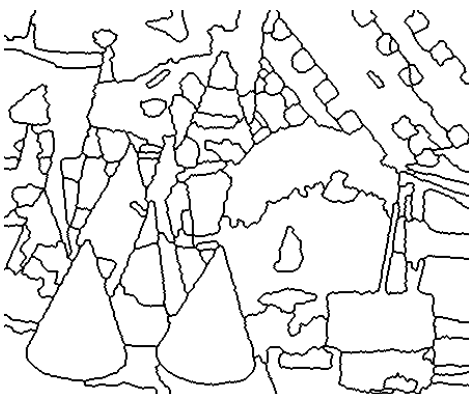
Before the fusion process, the dynamic range of depth data was matched to Y, again using the algorithm proposed in [90]. The multiscale transform adopted was the Gaussian pyramid; the feature selection rule was choosing the maximum of the two corresponding transform values. Figure 4.6 exhibits some selected results obtained with this method.



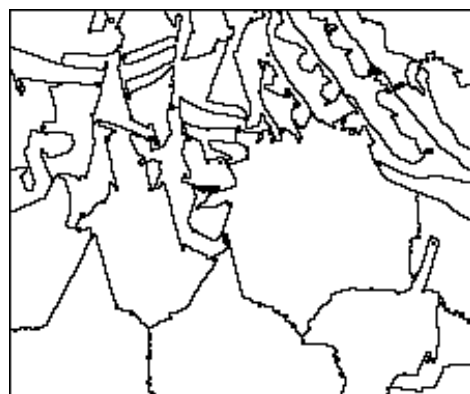
(a) Fused 'cones' image.



(b) Mean shift result. $d_{sym} = 66.33$, $d_{mut} = 5.07$



(c) JSEG result. $d_{sym} = 67.25$, $d_{mut} = 4.24$



(d) NCut result. $d_{sym} = 63.98$, $d_{mut} = 15.60$

Figure 4.6: Results for original algorithms with the multiresolution approach.

4.4 Discussion

The evaluation of the data in, data out model with the set of images with perfect depth information allows discarding this technique for such a job. The quality of the segmentations achieved with such approach is roughly the same as for the colour images alone. The algorithms are unable to correctly identify the main objects in the images, with parts of objects being grouped with the wrong object. Under the measures adopted this translates in achieving a high partition-distance d_{sym} (the algorithms are also unable to control the segmentation resolution), with a high mutual partition-distance d_{mut} , as the segmentations are not consistent. Because the results attained with the image data set with perfect depth information were unattractive, these techniques were not tested with the data set with noisy depth information.

Chapter 5

Data concatenation approach for hybrid image segmentation

The two techniques evaluated up until now allow the use of standard and established segmentation algorithms without modification by creating a new colour triplet containing information from the original colour and depth images. Although trivial to apply, the advantages were also found to be limited.

Another straightforward approach is just to concatenate the data, by juxtaposing all the data from the two images in an augmented vector, figure 5.1.



Figure 5.1: Data concatenation approach for hybrid image segmentation.

This procedure allows a joint modelling of all features in a compound fashion. However, modelling of such a vector may be extremely difficult due to the higher dimensionality, and complex statistical characteristics of disparate sources.

With this approach, conventional algorithms need to be adapted to deal with a fourth data channel.

The JSEG algorithm could be augmented with the depth information at different stages:

1. during the colour quantisation phase the incorporation of depth as a fourth ‘colour’

		‘chess’	‘billiards’	‘teacup’	‘cones’
original method	parameters	(7; 6.5; 1024)	(7; 6.5; 1024)	(7; 6.5; 1024)	(7; 6.5; 1024)
	regions	54	41	55	49
	d_{sym}	48.77	56.24	65.18	69.85
	d_{mut}	10.40	1.05	16.58	6.19
modified method	parameters	(14; 13; 13; 1024)	(14; 13; 13; 1024)	(14; 13; 13; 1024)	(14; 13; 13; 1024)
	regions	49	37	46	42
	d_{sym}	40.13	62.60	37.85	61.24
	d_{mut}	2.60	0.70	3.16	2.04

Table 5.1: Results for the Mean-Shift based algorithms over the test dataset with perfect depth information.

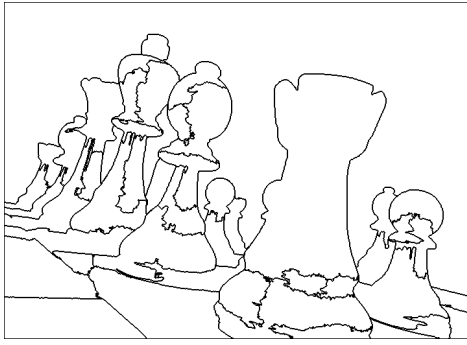
channel would allow keeping apart clusters of points with similar colour on spatially joint areas in the XY plane but with different depths. Note that once a set of points is incorporated in the same cluster they will not be separated again;

2. the J indicator is naturally extended to make use of the depth information as a third spatial dimension;
3. During the merging process, the quantized colours from the colour quantization process are used as colour histogram bins. The colour histogram for each region is extracted and the distance between two colour histograms i and j is calculated by $D_{CH}(i, j) = ||P_i - P_j||$, where P denotes the colour histogram vector. Again, the depth information could be used as a fourth colour channel, contributing to the discrimination of two histograms.

The NCut algorithm could also be modified by reflecting in the weight of the edges of the graph the depth difference between each pair of points.

However, this is even a more a natural extension to algorithms based on unsupervised clustering techniques from machine learning: the segmentation method based on the mean-shift algorithm is one of such kind. We adapted the publicly available software to receive a second image containing the depth information, adding a third parameter, the bandwidth for the depth data.

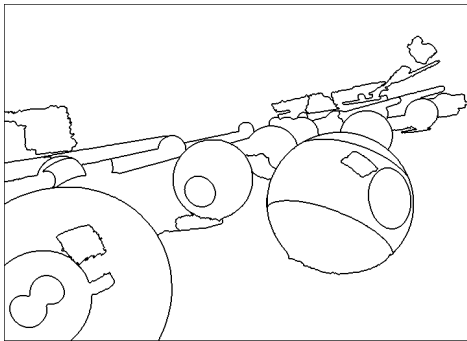
When applying the modified method to the test set with perfect depth maps, the results seem promising, revealing a significant improvement over the standard method (figure 5.2 and table 5.1). Note however that the extension of the algorithm favours a symmetric treatment for the colour and depth information. This may be inappropriate for real-life circumstances, where the depth information is noisier than colour information. As such, the assessment of this approach with the second test dataset was advised, and conducted with the results shown in figures 5.3 and table 5.2.



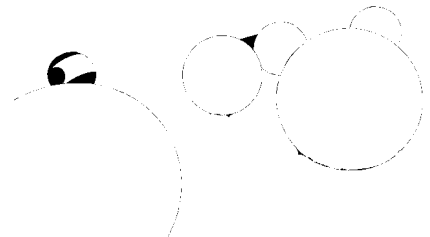
(a) 'chess' image.



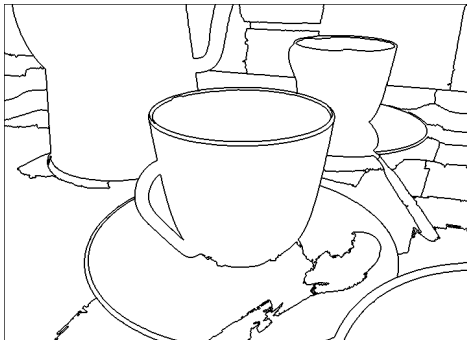
(b) 'chess' error mask.



(c) 'billiards' image.



(d) 'billiards' error mask.



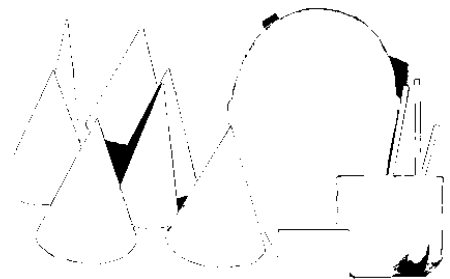
(e) 'teacup' image.



(f) 'teacup' error mask.



(g) 'cones' image.



(h) 'cones' error mask.

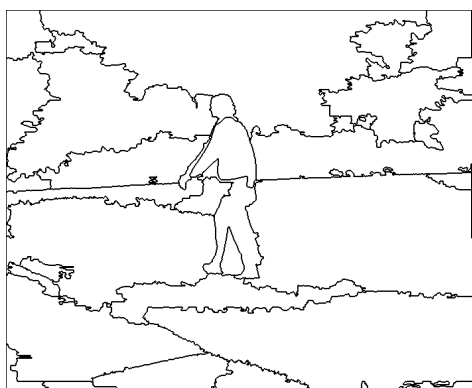
Figure 5.2: Results for the Mean-Shift Modified algorithm, over the test dataset with perfect depth information.



(a) 'walk street' image.



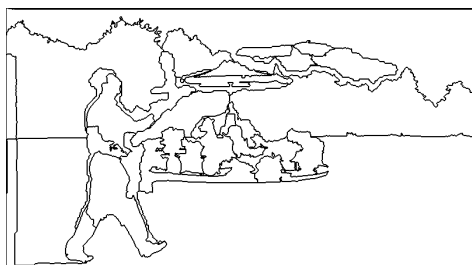
(b) 'walk street' error mask.



(c) 'walk park' image.



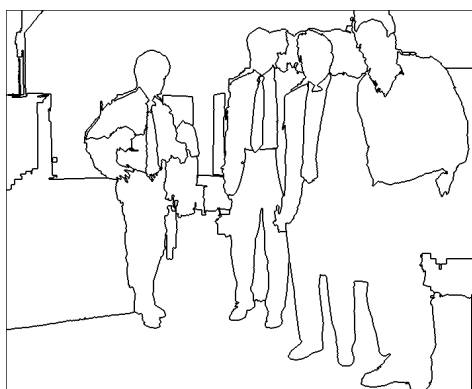
(d) 'walk park' error mask.



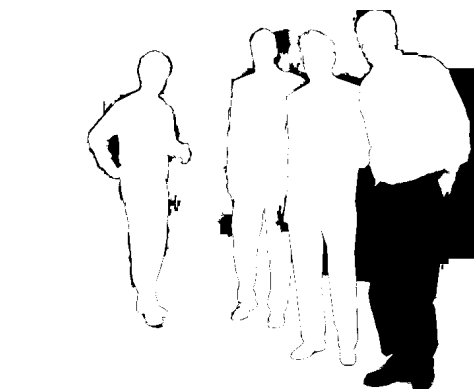
(e) 'juggler' image.



(f) 'juggler' error mask.



(g) 'men' image.



(h) 'men' error mask.

Figure 5.3: Results for the Mean-Shift Modified algorithm over the test dataset with noisy depth information.

		‘walk street’	‘walk park’	‘juggler’	‘men’
original method	parameters	(7; 6.5; 1024)	(5; 4; 1024)	(7; 6.5; 1024)	(7; 6.5; 1024)
	regions	40	49	26	50
	d_{sym}	37.64	29.78	25.54	66.12
	d_{mut}	4.91	2.92	4.32	9.46
modified method	parameters	(14; 13; 13; 1024)	(14; 13; 13; 1024)	(14; 13; 13; 1024)	(14; 13; 13; 1024)
	regions	44	32	33	39
	d_{sym}	40.13	62.60	37.85	61.24
	d_{mut}	6.99	4.16	3.85	12.99

Table 5.2: Results for the Mean-Shift based algorithms over the test dataset with noisy depth information.

The results attained for the dataset with noisy depth information expose two properties inherent to this approach:

- the joint, symmetric exploitation of colour and depth data allows a better identification of objects in the image: notice the heads, coherently segmented by the modified method, distinguished from the background;
- the noise present in the depth data translates into a lost of quality of the contours, becoming noisier, departing from their true location: observe the error involving each person in the images.

The main drawback just identified has its root in the symmetric use of colour and depth information. We present next a tentative solution to this problem.

5.1 Contour refinement

To retain the advantage of this method of correctly identifying the main regions present in the image but improve the quality of the edges, an additional stage was gauged: first, each region is eroded with a structuring element large enough to leave the influence zone of depth noise near edges — its size will depend on the noise level in the depth data; then, a region growing mechanism is performed using the colour information alone (or in conjunction with the depth data, but asymmetrically). A well-known choice for image segmentation based on region growing is the watershed algorithm [92]. That can be achieved by modifying the image so that it only has regional minima wherever the markers are nonzero, using the H-minima transform [93].

Performing a morphological erosion with a centered structuring element of size 17×17 for the noisy dataset, followed by a watershed using the eroded regions as markers, we attained the results reported in table 5.3. We tested different combinations for the gradient inputted to the watershed algorithm: first only a sum of the gradient of the three colour channels,

		‘walk street’	‘walk park’	‘juggler’	‘men’
colour gradient only	regions	48	42	40	43
	d_{sym}	39.03	61.85	37.95	61.44
	d_{mut}	7.17	4.18	3.91	13.25
summed	regions	47	44	40	43
	d_{sym}	38.95	61.82	38.18	60.35
	d_{mut}	6.85	4.51	3.89	12.98
modulated	regions	48	42	41	43
	d_{sym}	39.02	61.75	38.02	61.14
	d_{mut}	7.11	4.10	3.49	12.88

Table 5.3: Results for the modified mean-shift method over the test dataset with noisy depth information, with contours refined by region growing.



Figure 5.4: ‘walk park’ error image, with edge refinement with modulated gradient: error mask for d_{mut} .

in the CIE $L^*a^*b^*$ colour space; then a combination of the three colour gradients with information of the gradient information from the depth data. Being the noise of the depth data especially relevant near edges, the gradient of depth data was inputted as the difference between the maximum and minimum values of a depth over a window large enough to absorb the location uncertainty — this constitutes the morphological gradient of depth, $\mathcal{MG}(d)_{w \times h}$. We set the window to $w \times h = 17 \times 17$. Finally, instead of summing the depth gradient to the colour gradient, we also modulated the colour gradient by a measure of the depth gradient. Summarizing, three different gradients were fed to the watershed algorithm, to know:

- $\mathcal{G} = \mathcal{G}(L) + \mathcal{G}(a) + \mathcal{G}(b)$.
- $\mathcal{G} = \mathcal{G}(L) + \mathcal{G}(a) + \mathcal{G}(b) + \mathcal{MG}(d)_{w \times h}$.
- $\mathcal{G} = (\mathcal{G}(L) + \mathcal{G}(a) + \mathcal{G}(b)) e^{\alpha \mathcal{MG}(d)_{w \times h}}$. Currently $\alpha = 0.02$.

5.2 Discussion

The joint modulation of depth and colour information looks like the best approach for hybrid image segmentation, taking full advantage of all available information. However, the noise in the depth information needs to be conveniently handled. The symmetric use of depth and colour information in a modified well-known image segmentation method enabled to improve the identification of the main objects present in the image — observe the head of persons in the images. However, with it came a noisy identification of contours, motivated by the depth quality usually available in real-time scenarios — focus on the contour of the legs. Carrying on a refinement stage on contours allowed us to attain a superior performance.

Chapter 6

Hybrid image segmentation by fusion of decisions[§]

The attempts analysed so far to fuse depth and colour information have been symmetric with respect to the two sources (except for refinement step carried on the technique presented in the last chapter). However, colour and depth have different degrees of reliability, i.e., in practice depth information is noisier and with lower resolution than colour information. To account for this we could associate a “data set reliability” with each data set so that a less reliable dataset has less effect on the fusion process. Even if that was implemented — by modifying a standard segmentation algorithm — we were still left with the problem of estimating the number of segments in the image. Moreover, the possible misalignment between colour and depth information had also to be taken into account.

As such, we argue that a practical framework for hybrid image segmentation should

1. use the depth information to automatically estimate the number and localization of objects in the image. This process should produce markers (‘hints’) to guide the segmentation of the colour image. The colour information may be used together with the depth to assist this process.
2. perform a guided image segmentation, using *essentially* the colour information, starting from the markers previously created.

We will now investigate each of these two steps.

[§]Some portions of this chapter appeared in [49].

6.1 Automatic marker extraction

In most real-life images, objects have large vertical sections. In order to exploit this property for object segmentation, and as others before [94–96], a density image is defined by transforming the depth information on the XY plane to the XZ plane, where the value at position (x, z) of the density image denotes the number of points in the depth image at position x , taking the value z (by ‘integrating’ along the Y direction): let $D(x, y)$ be the depth information value at position (x, y) and $d(x, z)$ the value of the density image at position (x, z) ; then

$$d(x, z) = \sum_y \delta(D(x, y) - z)$$

$$\text{where } \delta(n) = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{otherwise} \end{cases}$$

While early efforts have exploited the XZ image segments to infer bounding boxes for objects in the XY image, our attempts provided limited results, as the extracted bounding boxes do not bound the objects completely, see figure 6.1.

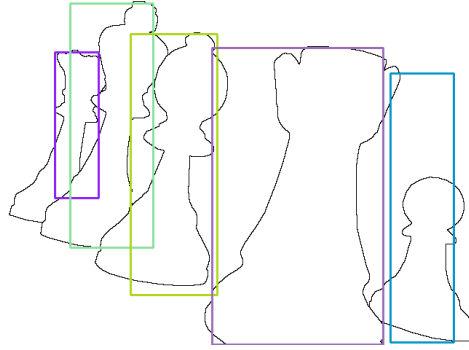


Figure 6.1: Bounding boxes for the ‘chess’ image.

This scenario suggested the use of the XZ image for object marker extraction; that can be accomplished using a simple threshold technique. More generally, we incorporated in this phase

1. a *pre-processing step*, which can include a low-pass filter, morphological operations, histogram equalisation or other preparatory operations. Our implementation performs a centered morphological opening operation, with a rectangular structuring element of size $(2o_h + 1) \times (2o_v + 1)$, followed by a centered morphological closing operation, with a rectangular structuring element of size $(2c_h + 1) \times (2c_v + 1)$;

2. an “*hysteresis*” *thresholding operation*. If a value is not inferior to the upper threshold limit, t_h , it is immediately accepted; if the value lies below the low threshold, t_l , it is immediately rejected; points which lie between the two limits are accepted if they are connected to pixels which exhibit strong response (at least t_h);
3. a *connected component analysis*. Each connected component is identified as an object marker. Our system uses 8-connected neighbourhoods;
4. a *post-processing step*, with objects with z values less than a predefined value (10% of the maximum possible z value in our implementation) being ignored. Low z values correspond to points farthest away of the camera or points to which the depth could not be estimated.

The resulting object segmentations, in the XZ image, for the test images, are presented in figures 6.2(a), 6.2(c), 6.2(e), 6.2(g), 6.3(a), 6.3(c), 6.3(c) and 6.3(g); each object is represented with a unique colour.

The object markers can be transported to the XY plane by including a pixel (x, y) in the marker of the object \mathcal{O}_i if the corresponding $(x, z) = (x, D(x, y))$ value in the density image lies in the object marker \mathcal{O}_i , figures 6.2(b), 6.2(d), 6.2(f), 6.2(h), 6.3(b), 6.3(d), 6.3(d) and 6.3(h).

While the pre-processing by morphological opening was implemented mainly as a kind of noise removal, with the effect of eliminating small and thin objects, the pre-processing by closing has the effect of filling small and thin holes in objects, and *connecting nearby objects*. This last property is of major importance in the presence of real-life depth maps. The quantisation effect in depth information, when sufficiently severe, is responsible for the formation of small vertical strips, incorrectly identified as individual objects (figure 6.4(a)). Morphological closing, with a structure element large enough to overcome the quantization effect on depth, allows to restore the object connectivity, figure 6.3(a).

6.1.1 Marker refinement

As visible in figure 6.2(b) markers may extend beyond the object or fill only a small part of the object. If the next step, the guided image segmentation, is a growing mechanism, markers extending beyond the objects are an issue. Therefore, a marker refinement step should be carried out to extend markers to pixels with high probability of belonging to the object and remove the markers from pixels with low probability. By simultaneously using the colour and depth information, markers can be extended to pixels where colour and depth information ‘agree’ and deleted from pixels where they ‘disagree’.

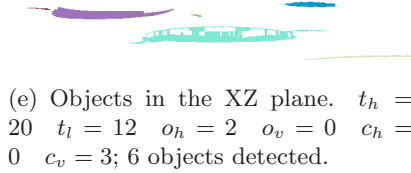
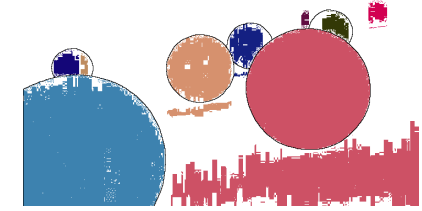
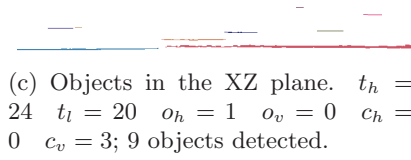
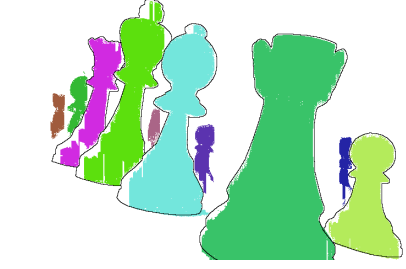
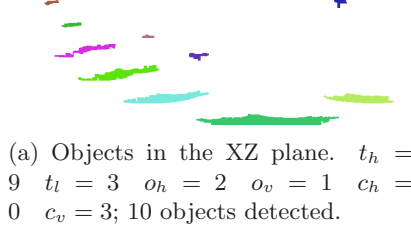
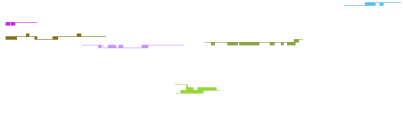


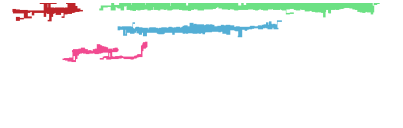
Figure 6.2: Automatic marker extraction for the test image set with perfect depth information.



(a) Objects in the XZ plane. $t_h = 40$ $t_l = 20$ $o_h = 2$ $o_v = 0$ $c_h = 0$ $c_v = 3$; 16 objects detected.



(c) Objects in the XZ plane. $t_h = 80$ $t_l = 40$ $o_h = 2$ $o_v = 0$ $c_h = 0$ $c_v = 3$; 7 objects detected.



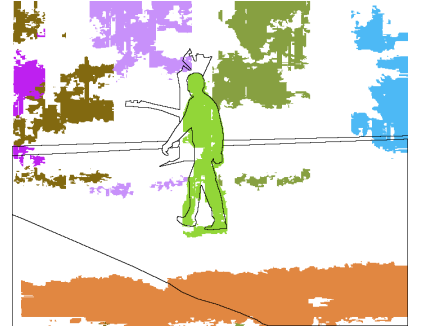
(e) Objects in the XZ plane. $t_h = 9$ $t_l = 3$ $o_h = 2$ $o_v = 1$ $c_h = 0$ $c_v = 3$; 4 objects detected.



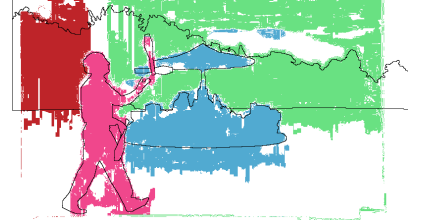
(g) Objects in the XZ plane. $t_h = 56$ $t_l = 16$ $o_h = 2$ $o_v = 0$ $c_h = 0$ $c_v = 2$; 8 objects detected.



(b) Markers in the XY plane, superimposed on the ground truth segmentation.



(d) Markers in the XY plane, superimposed on the ground truth segmentation.



(f) Markers in the XY plane, superimposed on the ground truth segmentation.



(h) Markers in the XY plane, superimposed on the ground truth segmentation.

Figure 6.3: Automatic marker extraction for the test image set with noisy depth information.

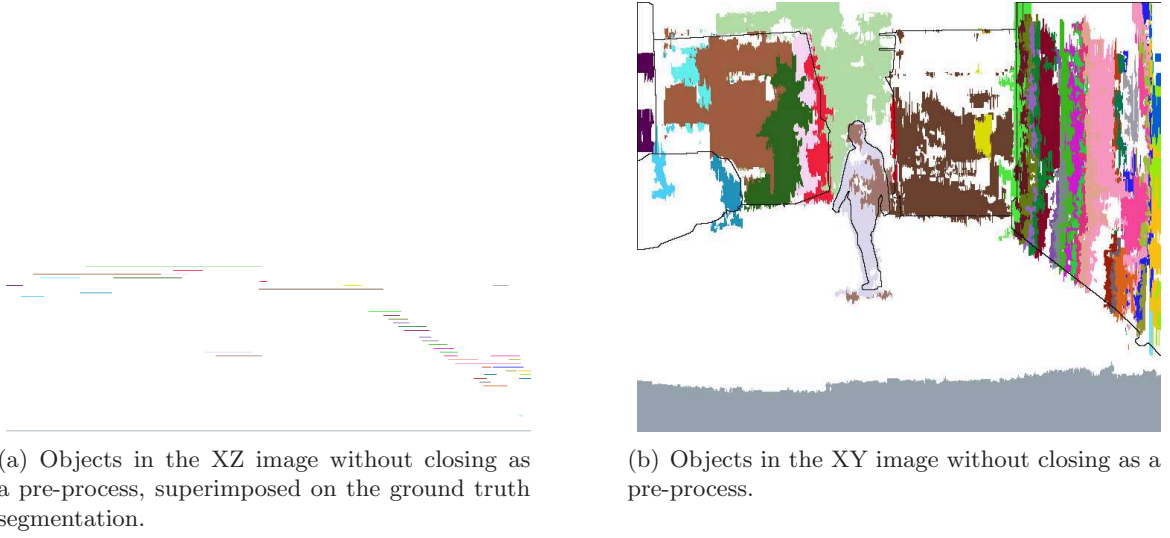


Figure 6.4: Importance of morphological closing as a pre-process.

The simplest approach is just to erode ‘enough’ the marker to remove pixels wrongly marked. More generally, an algorithm can be devised to learn to robustly differentiate objects from the marked pixels; then markers can be deleted from pixels with low confidence and enlarged to pixels with high probability.

A solution of this kind is to use the pixels marked in the first step to train a multiclass classifier to discriminate the objects based on six features: x, y, z and the colour triplet ($L^*a^*b^*$ in our implementation). Using the trained classifier, probabilities can be estimated for each pixel and a probability image and a class image map constructed. From the initial markers an average pixel probability is estimated for each object \mathcal{O}_i . Finally, pixels attributed to object \mathcal{O}_i but with probability below average are removed.

6.2 Guided image segmentation

Many approaches are possible for the segmentation using the hints produced in the previous stage. A well-known choice for guided image segmentation algorithm is based on the watershed [92]. That can be achieved by modifying the image so that it only has regional minima wherever the markers are nonzero, using the H-minima transform [93].

This technique needs a marker to the outer of the detected objects. Toward that end the distance to the closest marked pixel is computed for all pixels — marked pixels are naturally at 0 distance. Because it is likely that object markers do not fill completely the corresponding object, the outside marker was defined by keeping only the pixels at greatest distance (we

kept pixels at distance of at least 0.25 of the maximum value).

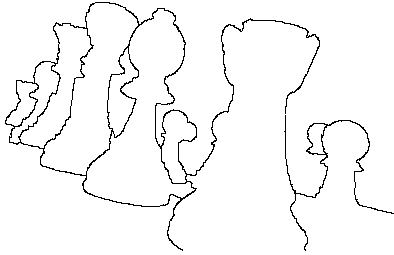
Next, objects' markers were refined as previously explained using a morphological erosion — erosion element of size 17×17 , centered at $(8, 0)$. The asymmetry adopted for the vertical direction is explained by the main cause for markers extending outside the objects: objects in contact with the ground. Before erosion, each marker was pre-processed by filling possible holes in it; after erosion, and because it is likely that both the transportation of the markers from the XZ image to the XY image and the erosion step lead to disconnected markers, we keep only the largest connected component, for each marker.

Colour- and intensity-gradient information are combined to obtain a final gradient capturing all perceptual edges in the image. It would be desirable to also incorporate the depth information in the final gradient, with the aim to leave it unmodified in areas at the same depth but emphasize in areas of different depths.

We tested different combinations for the gradient inputted to the watershed algorithm: first only a sum of the gradient of the three colour channels, in the CIE L*a*b* colour space; then a combination of the three colour gradients with information of the gradient information from the depth data. Being the noise of the depth data especially relevant near edges, the gradient of depth data was inputted as the difference between the maximum and minimum values of a depth over a window large enough to absorb local uncertainty — this constitutes the morphological gradient of depth, $\mathcal{MG}(d)_{w \times h}$. We set the window to $w \times h = 5 \times 5$ for the dataset with perfect depth and $w \times h = 17 \times 17$ for the dataset with noisy depth data. Finally, instead of summing the depth gradient to the colour gradient, we also modulated the colour gradient by a measure of the depth gradient. Summarizing, three different gradients were fed to the watershed algorithm:

- $\mathcal{G}_1 = \mathcal{G}(L) + \mathcal{G}(a) + \mathcal{G}(b)$.
- $\mathcal{G}_2 = \mathcal{G}(L) + \mathcal{G}(a) + \mathcal{G}(b) + \mathcal{MG}(d)_{w \times h}$.
- $\mathcal{G}_3 = (\mathcal{G}(L) + \mathcal{G}(a) + \mathcal{G}(b)) e^{\alpha \mathcal{MG}(d)_{w \times h}}$. Currently $\alpha = 0.02$.

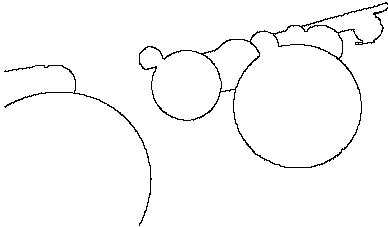
Results are presented in table 6.2; for each image, the example with the best performance was selected for visual presentation in figure 6.5. Appreciate the good control of the number of segments achieved, without compromising the quality of the segmentation, a side effect for conventional methods. The objects were correctly identified as a whole; under the measures adopted, this translates in achieving a low partition-distance, d_{sym} (and a low number of regions, R), without affecting the consistency of the segmentation, as measured by d_{mut} . Confronting with results from conventional methods, as presented in tables 1.1 and 1.2, and figures 1.6, 1.7, 1.8, 1.9, 1.10 and 1.11, we observe clear improvements: the almost meaningless results yielded by conventional methods for the 'chess' image are transformed in



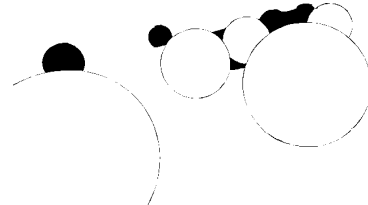
(a) 'chess' segmentation image.



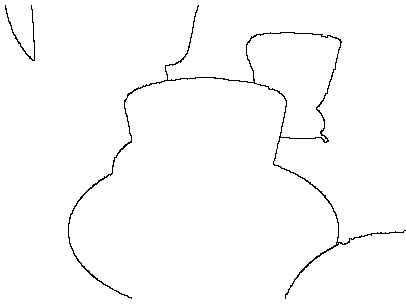
(b) 'chess' error image.



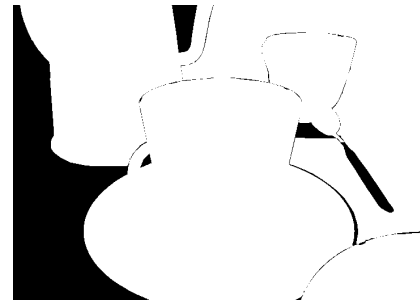
(c) 'billiards' segmentation image.



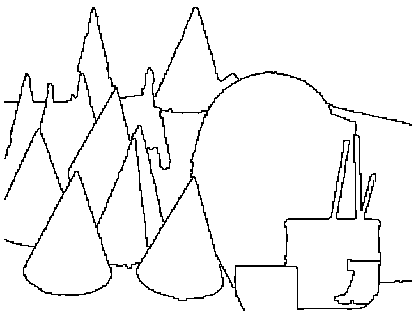
(d) 'billiards' error image.



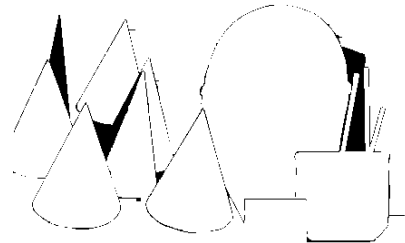
(e) 'teacup' segmentation image.



(f) 'teacup' error image.

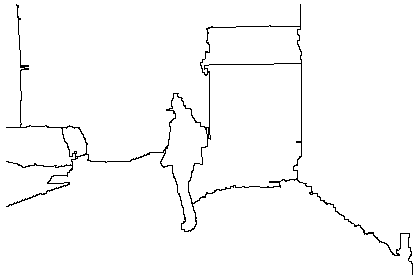


(g) 'cones' segmentation image.

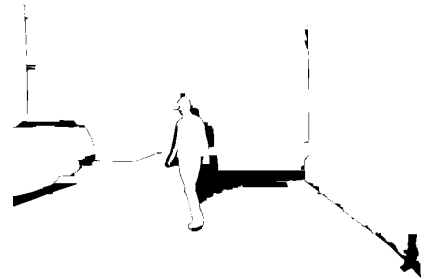


(h) 'cones' error image.

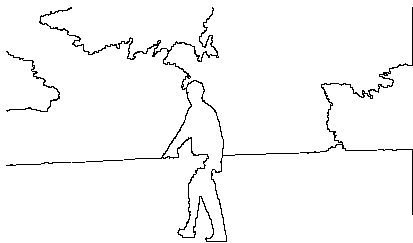
Figure 6.5: Results for the watershed with markers algorithm over the test dataset.



(a) 'walk street' segmentation image.



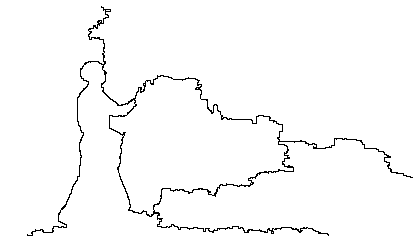
(b) 'walk street' error image.



(c) 'walk park' segmentation image.



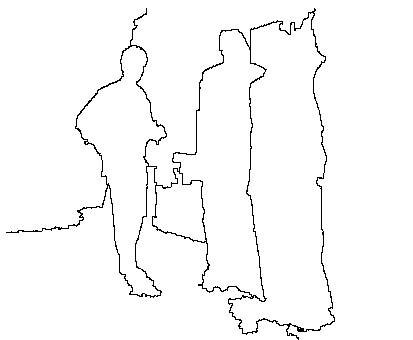
(d) 'walk park' error image.



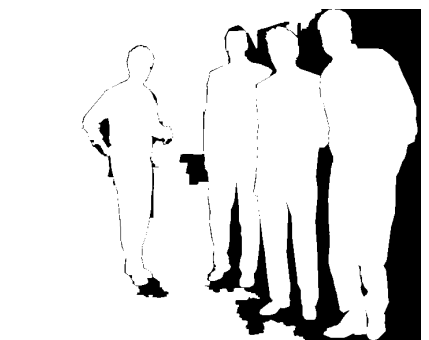
(e) 'juggler' segmentation image.



(f) 'juggler' error image.



(g) 'men' segmentation image.



(h) 'men' error image.

Figure 6.6: Results for the watershed with markers algorithm over the test dataset.

		'chess'	'billiards'	'teacup'	'cones'
\mathcal{G}_1	regions	11	8	6	20
	d_{sym}	11.18	27.28	39.93	29.95
	d_{mut}	8.97	11.58	18.67	6.76
\mathcal{G}_2	regions	11	8	6	20
	d_{sym}	8.25	3.48	37.75	28.39
	d_{mut}	3.80	2.32	16.46	3.43
\mathcal{G}_3	regions	11	8	6	20
	d_{sym}	10.06	28.36	38.11	28.30
	d_{mut}	7.17	12.59	16.86	4.57

Table 6.1: Results for the watershed with markers method over the test set with perfect depth information.

		'walk street'	'walk park'	'juggler'	'men'
\mathcal{G}_1	regions	10	7	5	8
	d_{sym}	29.05	42.80	45.14	42.36
	d_{mut}	9.60	16.26	31.89	19.46
\mathcal{G}_2	regions	10	7	5	8
	d_{sym}	14.55	42.36	44.58	42.11
	d_{mut}	3.40	20.94	34.61	15.92
\mathcal{G}_3	regions	10	7	5	8
	d_{sym}	15.94	36.57	45.65	40.52
	d_{mut}	4.19	3.05	28.64	17.67

Table 6.2: Results for the watershed with markers method over the test set with noisy depth information.

a high-quality segmentation; the over-segmentations produced by the mean shift algorithm or the difficulty of the JSEG algorithm to separate the man from its surroundings, in the 'walk street' and 'walk park' images, gives rise to an essentially foreground / background segmentation, with the man correctly isolated from the background. Naturally these results should not be compared with those from methods inferring the background from an image sequence, using temporal information.

We also observe that the incorporation of the depth information in the gradient fed to the watershed algorithm leads to important gains in terms of the quality of the final segmentation.

6.3 Discussion

This chapter presents a new approach to image segmentation. The main idea is to use depth information to guide a segmentation using essentially the colour information. This method is likely to produce simpler segmentations, less over-segmented, and compares favourably with state-of-the-art methods. The use of active contours in the guided segmentation phase may further improve the results, with a superior performance when markers extend beyond the objects. A solution of this kind eliminates the need for a refinement operation between the marker extraction and the guided image segmentation stages, due to its ability to expand or contract as appropriate.

Chapter 7

Image segmentation assisted by depth and motion information

We have tackled the issue of improving the quality of image segmentations with depth information. This can help overcome problems such as over-segmentation, and the extra information can increase the robustness and accuracy of the segmentation. However, depth is not the Graal of the segmentation problem. Consider the case when two objects moving in opposite directions cross with each other. If their depth is similar, even the methods presented so far will have problems to divide the two objects. Or, as already observed, it is still difficult to segment objects from the ground where they stand. This creates the interest for integrating more information in the segmentation techniques.

In this chapter we will concentrate on extending the methods analysed so far to incorporate motion information in the segmentation process. As depth, motion is useful as a cue for image segmentation. Velocity information may be used to link adjacent but visually dissimilar surfaces or to divide surfaces not easily separable by static criteria alone. Often, ambiguous object boundaries in a single image frame are easily resolved when dynamic effects are evaluated based on a sequence of frames.

An image sequence is a series of two-dimensional images that are sequentially ordered in time. They can be acquired by video or motion picture cameras, or generated by computer graphics and animation techniques. The analysis of image motion and the processing of image sequences using motion information is becoming more and more important as video and television systems are finding an increasing number of applications in the areas of entertainment (motion pictures, HDTV), robot vision (autonomous navigation), education, personal communications (videophone), and multimedia.

There is some literature on systems for segmenting from motion. A common class of methods

for segmentation from motion is based on matching features points, such as corners or interest points. Since these systems process only a relatively sparse set of feature points, they are used to detect and track moving objects in a scene, rather than segmenting them with high resolution. Instead of matching feature points, some systems match small image blocks. These systems are preferably used in the context of low-bit-rate video coding. This method again results in a rather crude segmentation with a resolution given by the block-size. However, the purpose of video coding is in any case compression, rather than segmentation. Others, focusing on the simultaneous solution of motion estimation and segmentation assume a fixed number of regions and are still more concerned with motion estimation for compression [97]. The segmentation method presented here differs from these approaches as it focuses on the segmentation itself.

Another common class of system for segmenting from motion only try to split the scene into foreground and background regions. Many of these methods attempt to model on a per-pixel basis the background features as coming from a Gaussian distribution [98] or, in more sophisticated models, as a mixture of Gaussians [99] and non-parametric models [100]. When evaluating the current pixel, the recent history of observation is used to estimate the probability of the pixel belonging to the background. By allowing the background model statistics to be updated over time, methods gain robustness against slow background variations, such as illumination changes, addition or removal of objects, etc. By simultaneously taking advantage of colour and depth information, available in many systems, algorithms are able to perform reasonably well under more severe conditions [101]. These systems generally have a fairly long period to adapt and tend to fail catastrophically when confronted with a variety of real-world phenomena such as camera motion. Moreover, by modelling each pixel independently, it is difficult to maintain a spatial coherence in the segmentation.

Here we will integrate depth and motion for high quality image segmentation. If it is true that for synthetic sequences motion values can be computed exactly, that is not the typical scenario, where motion is *estimated* from a sequence of images. Then, our approach should be robust against inaccuracies in the motion information, as it is against in the depth information. A key observation when addressing the problem of segmenting assisted both by depth and motion information is that these two cases of distinct information, which are often treated separately, have in fact much in common (figure 7.1):

- depth information is typically computed from stereo information, with two images acquired simultaneously.
- motion information is typically computed from sequential information, with two images acquired sequentially.

Akin to motion techniques, a class of stereo methods is based on the matching of small blocks,

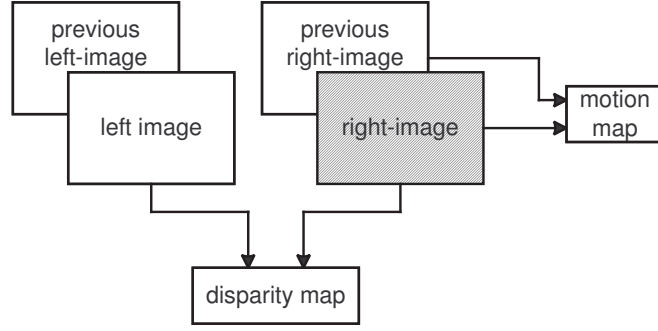


Figure 7.1: Motion and depth estimation for the segmentation of the right image.

figure 7.2. Therefore, techniques integrating depth and motion in the segmentation process should be symmetric with respect to these two sources. Let us analyse the consequences for the two major segmentation techniques already extended for depth information.

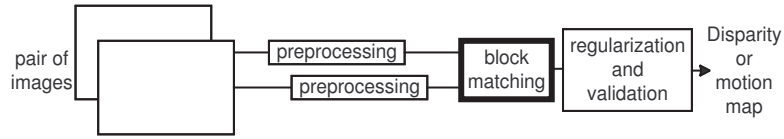


Figure 7.2: Matching algorithm behaviour for stereo algorithms.

7.1 Motion and depth assisted image segmentation with the Mean Shift algorithm

The mean shift algorithm, a nonparametric procedure for the analysis of multimodal data, performs naturally with multidimensional data; as such, its extension to integrate motion information is just a matter of adding a proper normalizing constant for motion data, h_v .

Along with its simplicity comes what can be a major drawback: now we may be feeding two noisy data channels to the algorithm, which may lead to unacceptable results. Previously, our only source of noise was the depth information, which was handled by taking an extra refinement step of borders. Now, that same procedure may reveal itself insufficient to recover from the errors introduced by two noisy data channels.

7.2 Image segmentation guided by noisy metadata

In the marker based approach, introduced in chapter 6, depth information was used to automatically estimate the number and localization of objects in the image. This process was conducted in an image density function, where the pixel value denotes the number of points in the depth image at pixel position. The higher the value, the higher the probability of that pixel belonging to an object. That value may be interpreted as a degree of membership to the foreground. Due to the similarity of depth and motion information, the motion can be tentatively integrated in the framework in the same way. From the motion vectors create a density image; then, it is left the problem of using two density images (obtained from depth and motion) to create markers. One approach is to produce a single density map, integrating both. Fuzzy logic [102,103] would seem to be the right tool.

Remember that the framework introduced in chapter 6 suggested to perform the segmentation in a two-step operation, with the first one comprising the use of depth information to create a depth density image, $d(x, z)$, from the depth-image, $Z(x, y)$, from where markers were extracted. These, in turn, were used to guide the colour segmentation in the second step of the proposed framework. Now, in the extended framework with motion, we could be driven to create a motion density image, $d(x, v)$, from the motion-image, $V(x, y)$, and combine it with the depth density image using some pre-selected fuzzy operation. Then we would proceed as before, extracting markers from this combined density image. However, note that $d(x, z)$ and $d(x, v)$ are defined over different domains. That hinders the direct merging of both densities. To surmount this problem, the integration of densities could be performed in the (x, y) plane, by first transporting both densities to this plane:

$$\begin{aligned} d_{xz}(x, y) &= d(x, z) & \text{if } Z(x, y) = z \\ d_{xv}(x, y) &= d(x, v) & \text{if } V(x, y) = v \end{aligned} \tag{7.1}$$

Now we could be tempted to perform the selected fuzzy operation on these two density-images. In spite of the effort this approach will not be fruitful. Consider the image 7.3(a) and the corresponding depth- and motion-maps in figures 7.3(b) and 7.3(c). Two of the three cubes are moving in opposite directions; the leftmost cube is at rest. Proceeding as described in chapter 6, we can compute the density in the XZ plane (figure 7.4(a)) and in the XV plane (figure 7.5(a)). Transporting the density to the XY plane according to (7.1), we get figures 7.4(c) and 7.5(c). Note that $d_{xz}(x, y)$ attains approximately the same (high) value in all true cubes, with the two leftmost cubes ‘spatially merged’. Following the same reasoning, the $d_{xv}(x, y)$ density will not distinguish among the cubes. Operating with these two densities (interpreted as a fuzzy membership on the foreground ‘set’) it will not be possible to disconnect the cubes.

Continuing the quest for a workable solution, it seems that the path leads to working with

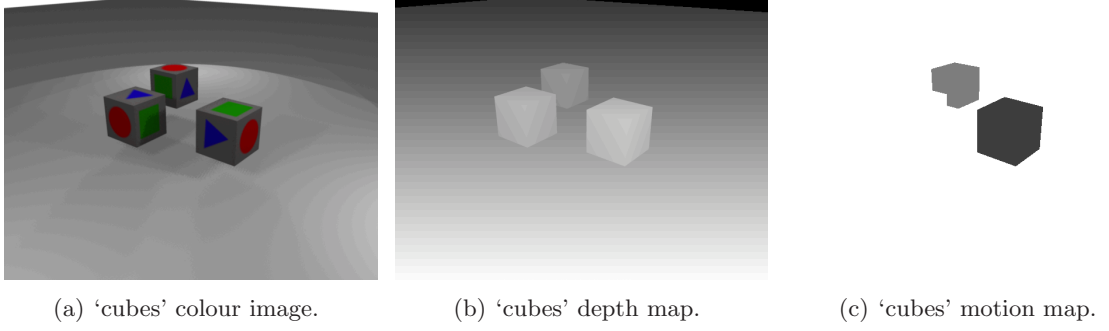


Figure 7.3: 'cubes' image.

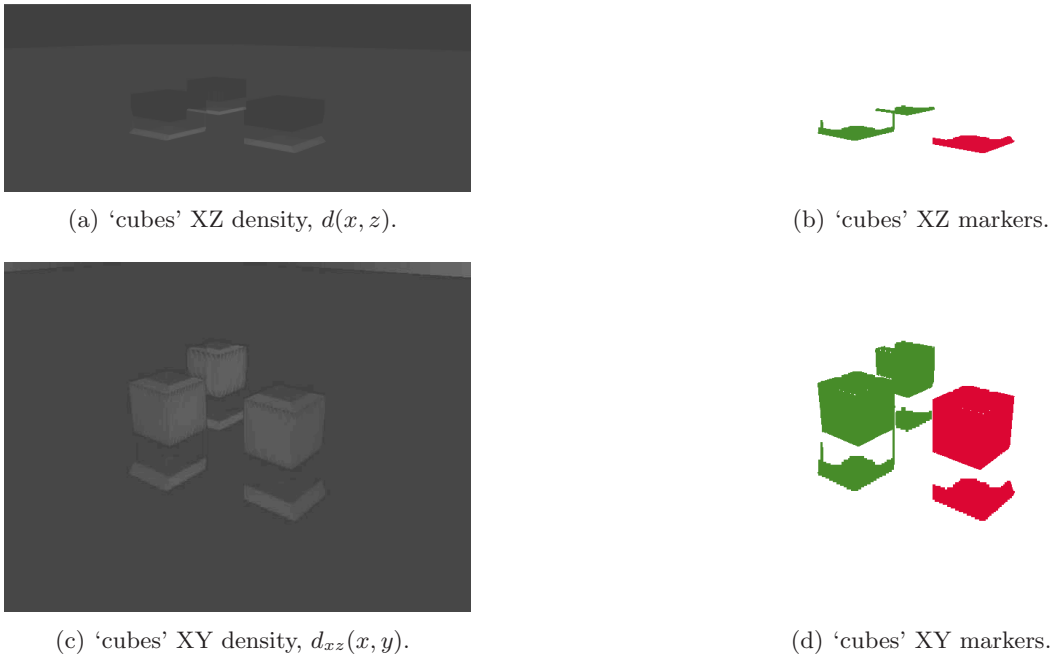


Figure 7.4: Depth density-images and corresponding markers.

a membership function per object. These could be estimated by taking the density value inside the object markers (figures 7.4(d) and 7.5(d)) and 0 (or some estimated value) outside. Then every object membership function in $d_{xz}(x, y)$ would be operated with every object membership function in $d_{xy}(x, y)$. Finally, results would have to be merged and validated. Although feasible, this is becoming an awkward and unmanageable solution. Let us restate our (simplified) goal. Having depth-markers and motion-markers, we want them to cross-validate each other and to allow depth-markers to sometimes divide motion-markers (for objects with similar movement at different depths) and the opposite, motion-markers to divide depth-markers (to divide objects at similar depth but with different movements). It does not take long to suspect that the already introduced intersection-graph between both

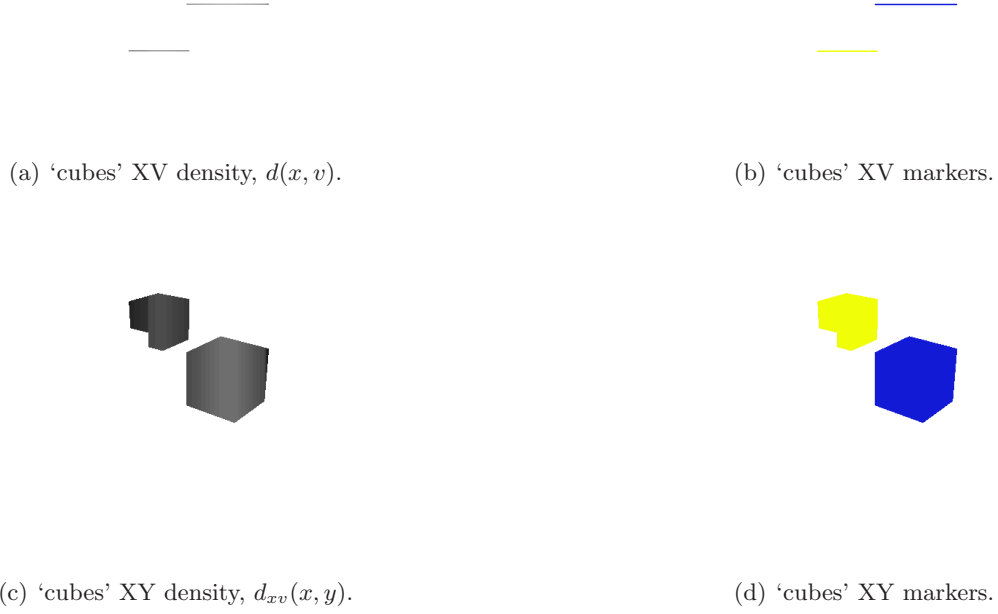


Figure 7.5: Motion density-images and corresponding markers.

maps provide a clean picture of every possible marker intersection, with a valuable insight into the solution.

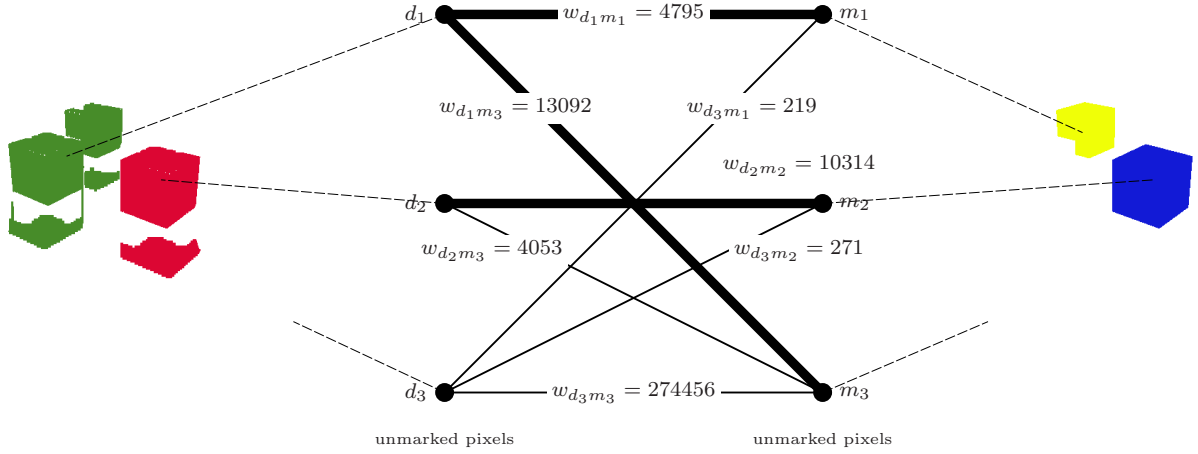
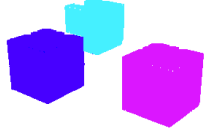


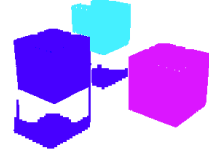
Figure 7.6: Intersection-graph for depth and motion maps.

In figure 7.6 we have represented the intersection-graph for the 'cubes' images. Here each node d_i represented the set of pixels belonging to the same marker in the depth map (with an extra node, d_3 , to represent the unmarked pixels); each node m_i represented the set of

pixels belonging to the same marker in the motion map (with an extra node, m_3 , to represent the unmarked pixels); and the weight of each edge represents the number of pixels in the intersection of a marker in the depth map with a marker in the motion map. We would like to come up with a sensible procedure yielding three markers in this example, represented thicker in figure 7.6, translating into the markers shown in figure 7.7(a).



(a) Ideal merged markers.



(b) Real merged markers.

Figure 7.7: Merging result for the ‘cubes’ image.

The problem now is to define a procedure of choosing which intersection will give rise to a new independent marker, and which will be aggregated under the unmarked pixels.

The unsuccessful search for a convenient global measure — such as the partition-distances already introduced or the maximum spanning tree — as a possibly interesting formalization of the marker fusion process, led us to adopt a local measure: we opt for associating a new marker to an intersection if the intersection weight is a substantial part of any of the two incident nodes (markers)[†]. Mathematically

$$\text{if } \max \left(\frac{w_{d_i, m_j}}{\sum_{\ell} w_{d_{\ell}, m_j}}, \frac{w_{d_i, m_j}}{\sum_{\ell} w_{d_i, m_{\ell}}} \right) \begin{cases} > \epsilon \text{ mark intersection } d_i, m_j \\ \leq \epsilon \text{ unmark intersection } d_i, m_j \end{cases}$$

Stated equivalently

$$\text{if } \min \left(\frac{\sum_{\ell} w_{d_{\ell}, m_j} - w_{d_i, m_j}}{\sum_{\ell} w_{d_{\ell}, m_j}}, \frac{\sum_{\ell} w_{d_{\ell}, m_j} - w_{d_i, m_j}}{\sum_{\ell} w_{d_i, m_{\ell}}} \right) \begin{cases} < (1 - \epsilon) \text{ mark intersection } d_i, m_j \\ \geq (1 - \epsilon) \text{ unmark intersection } d_i, m_j \end{cases}$$

Comparing the above measure with the Berkeley measures for comparing segmentations presented in chapter 2, and their mapping in the intersection-graph presented in (2.1), we

[†]If the edge is connecting a node corresponding to a marker with a node corresponding to unmarked pixels, the test should be made only with the node corresponding to the marker. If an edge is connecting two nodes corresponding to unmarked pixels, it is always unmarked.

conclude that both are based on the same local refinement error.

Adopting this procedure, the fused marker would yield ($\epsilon = 0.4$) as represented in figure 7.7(b). Note that three markers were indeed created. However, the noise present in the depth markers was not completely removed. In fact, because this noise is jointed in edge $w_{d_1 m_3}$ with pixels corresponding to the leftmost cube, it impossible to recover from it with this framework (without losing the marker of the leftmost cube). Even adopting a generic approach based on fuzzy logic, it would have been difficult to eliminate this noise, as the density values at the leftmost cube and at the noisy pixels is essentially the same.

We propose then to extend the technique presented in chapter 6 in the following way:

- create the XZ density image, operate on it to extract depth marker and transport them to the XY plane, as in the basic proposal.
- repeat the above procedure, now using the motion information, resulting in a motion marker image in the XY plane.
- merge both XY marker-maps using the local refinement error to prune markers in non-concordant pixels.
- apply the colour image segmentation guided by the fused markers.

Although we have implicitly assumed throughout this discussion that the motion information is in the scalar form, yielding a scalar motion map, that is not typically the case, with motion information available in the X and Y directions or in any other equivalent form such as (intensity, angle). In this case we would have two density images, from which two motion marker-images could be created and merged with the depth marker-image. Observing that, although the local refinement error is commutative it is not associative, the above-defined procedure would have to be conveniently extended to handle three or more marker-images. A possibility would be to generalize the local refinement itself to three or more images, similarly to [35]. Because in this work we will restrict to one motion marker-image, this generalization will not be further considered here.

7.3 Results with synthetic material

We have set up experiments using synthetic images to evaluate the methods proposed here. For our purposes we created a synthetic image sequence that has three target cubes in the center surrounded by a background, see figure 7.8. The sequence was rendered with Maya 7; the depth maps were generated using Maya's renderer; the motion maps were manually generated.

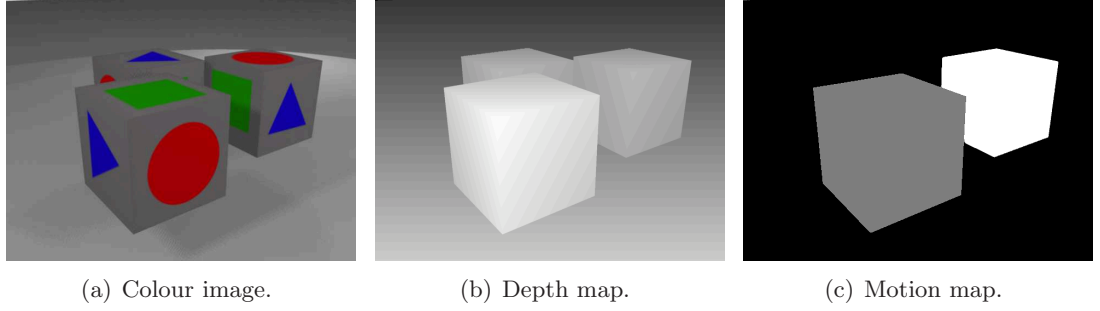


Figure 7.8: Frame 6 of 'cubes' sequence.

Following the procedure already formulated in chapter 6, depth markers in the XZ plane and motion markers in the XV plane were generated, as depicted in figure 7.9.

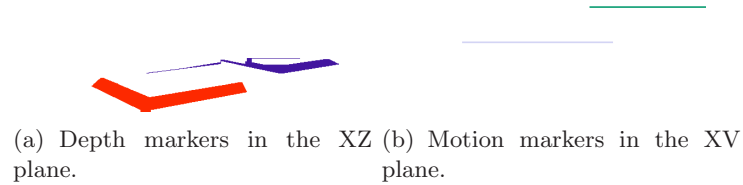


Figure 7.9: Extracted markers for frame 6 of 'cubes' sequence.

Then, markers were transported to the XY plane and merged following the method adopted in the previous section, creating three different marker images (fig:newcubesXYmarkers). Finally, the comparisons were done using three versions of the guided watershed segmentation: starting from depth markers we obtain segmentation 7.11(a), starting from motion markers it is attained the segmentation 7.11(b); from the merged markers results the segmentation 7.11(c). To be noticed is that the quality of the segmentation is clearly improved when we integrate motion information. When both depth and motion information is used to extract markers we are able to correctly divide the three cubes (although with a extra spurious region as side effect).

Finally, the same sustained improvement was detected with the mean shift based algorithms,

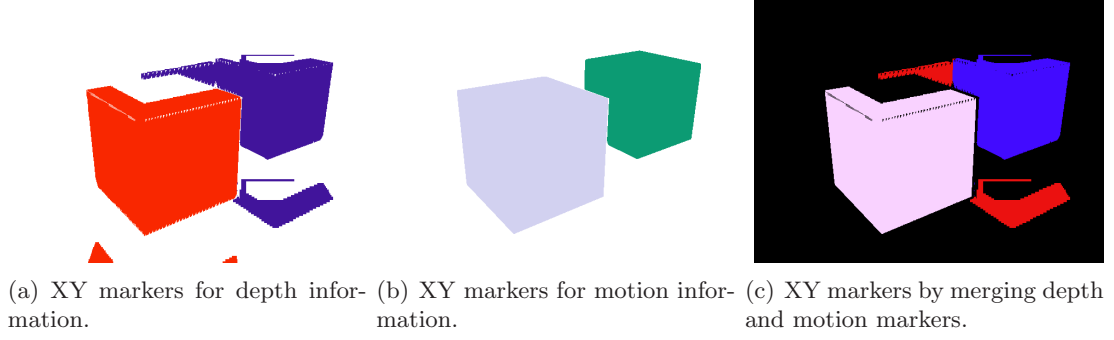


Figure 7.10: Extracted markers in the XY plane, for frame 6 of ‘cubes’ sequence.

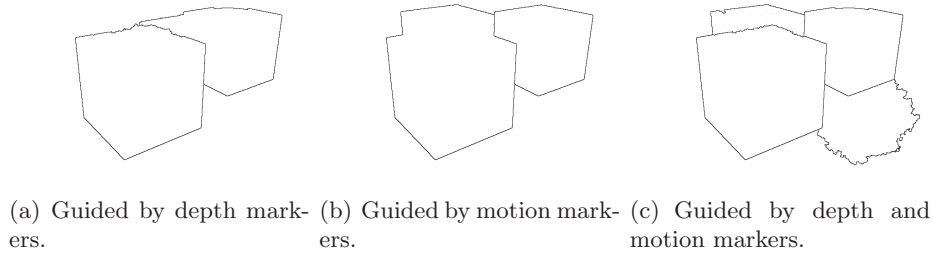


Figure 7.11: Watershed image segmentation guided by the extracted markers.

from the original algorithm based only on colour (figure 7.12(a)), to the modified versions, integrating depth (figure 7.12(b)), motion (figure 7.12(c)), and both (figure 7.12(d)).

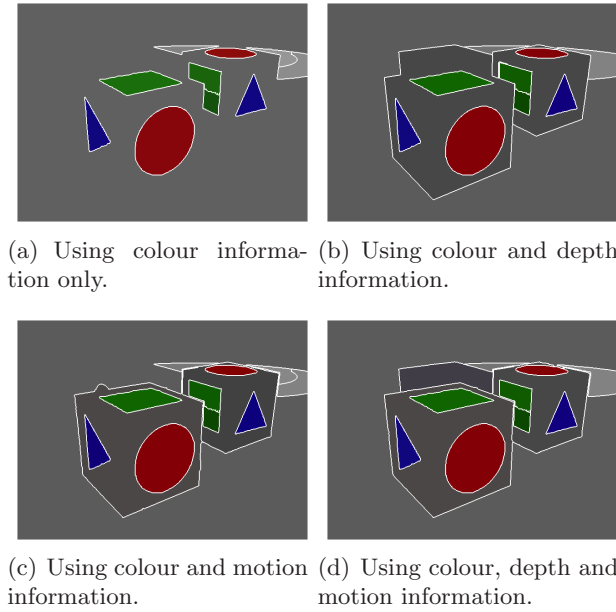


Figure 7.12: Segmentation of frame 6 of the ‘cubes’ sequence by mean shift based algorithms.

7.4 Results with real-life material

This section provides experimental results obtained on two stereo sequences acquired with a monochrome MEGA-D digital stereo head (by Videre Design) equipped with a pair of 4.8 mm lenses.[‡]Image size is 640×480 .

7.4.1 The ‘Outdoor’ sequence

Two temporally consecutive stereo pairs of the sequence is shown in figure 7.13.



(a) Left frame 218.



(b) Right frame 218.



(c) Left frame 219.



(d) Right frame 219.

Figure 7.13: ‘Outdoor’ stereo sequence.

The depth information, in the form of a disparity map, obtained with the Single Matching Phase (SMP) stereo algorithm [104], is also freely available at <http://labvisionone.deis.unibo.it/~smattoccia/stereo.htm>, from where it was downloaded. It is depicted in image 7.14(a) for frame 219.

The motion information was computed with a basic block matching algorithm, as implemented in the OpenCV software, with a block size of 16×16 and a search region of 65×65 . The obtained motion information in the X -direction is depicted in figure 7.14(b).

[‡]The sequences were downloaded from <http://labvisionone.deis.unibo.it/~smattoccia/stereo.htm>.

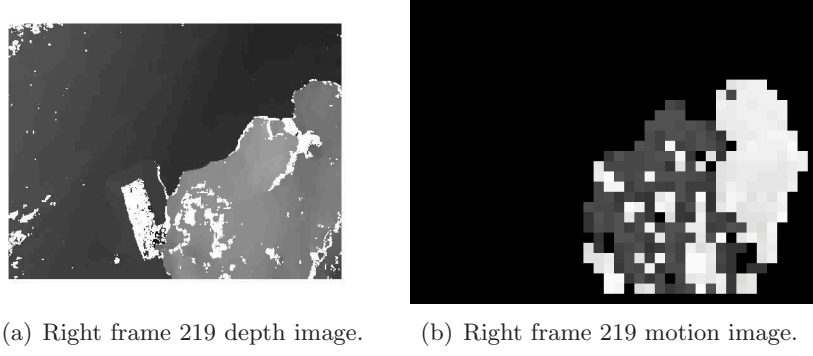


Figure 7.14: Computed depth and motion images for right frame 219 of ‘Outdoor’ sequence.

Note that the depth map is smaller than the original image (due to the stereo depth algorithm). Depth and motion markers were extracted (figures 7.15(a) and 7.15(b)) and transported into the XY plane, figures 7.15(c) and 7.15(d). We considered only the X component of the motion vectors. Observe that, unlike the motion information, the depth information was unable to create distinct markers for the two persons.

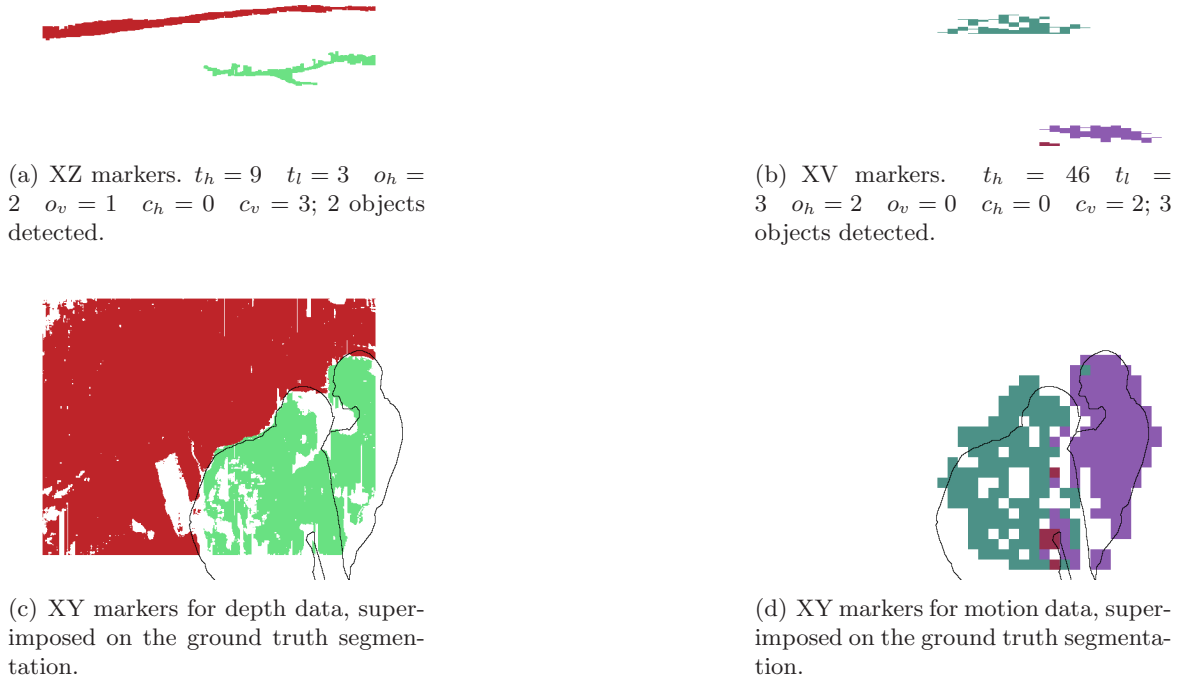


Figure 7.15: Extracted markers for right frame 219 of ‘Outdoor’ sequence.

Finally, we proceeded with the guided image segmentation step, experimenting three different possibilities for the initial markers: depth-markers only, motion-markers only, and the merging of depth- and motion-markers. Depth and motion markers were merged with

	D	M	DM	mean shift	mean shift D	mean shift M	mean shift DM
regions	2	3	5	14	38	22	38
d_{sym}	15.84	12.11	17.54	30.16	68.02	21.14	73.72
d_{mut}	9.64	7.02	6.65	13.75	9.00	6.50	3.75

Table 7.1: Results for frame 219 of the ‘Outdoor’ sequence. D – depth assisted; M – motion assisted; DM – depth and motion assisted.

$\epsilon = 0.4$. Due to the high level of noise present in the depth and motion information, only the colour information was used to compute the gradient fed to the watershed algorithm. Results are depicted in figure 7.16 and summarized in table 7.1. It is clear the advantage of integrating motion information in the marker extraction process. As also noted previously in the synthetic example, here too the integration of motion information leads to a decent division of the two men present in the image. It is important to stress that this was achieved with very low quality depth and motion information.

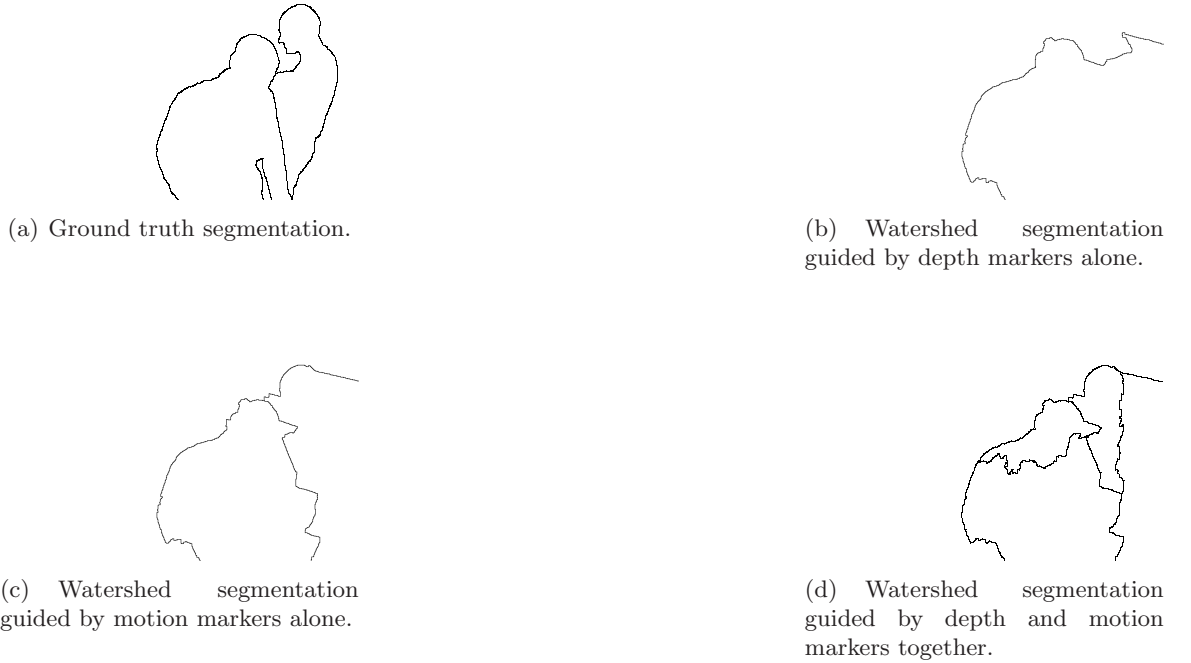
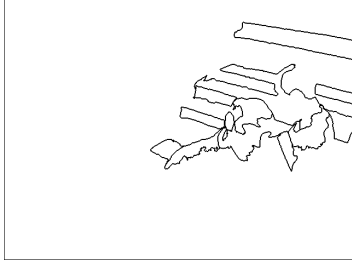
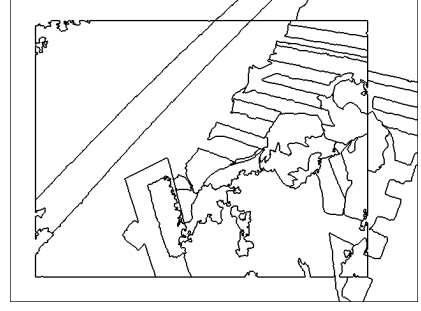


Figure 7.16: Results for frame 219 of the ‘Outdoor’ sequence.

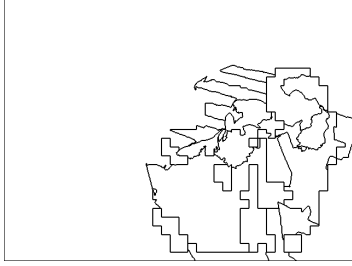
The standard mean-shift segmentation algorithm, as well as the enhanced modifications proposed in this work, was also applied to the ‘Outdoor’ sequence. The attained results are shown in figure 7.17 and in table 7.1. Here we observe that the strong presence of noise in the auxiliary information is having unacceptable consequences in the result, with the algorithm unable to recover from such noise. The frame around the segmentation using depth



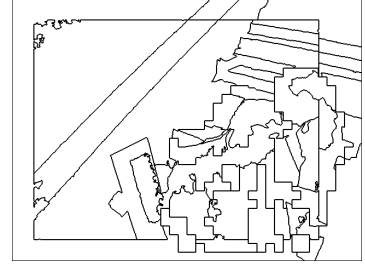
(a) Original mean shift algorithm using colour information only.



(b) Modified mean shift algorithm using colour and depth information.



(c) Modified mean shift algorithm using colour and motion information.



(d) Modified mean shift algorithm using colour, depth and motion information.

Figure 7.17: Results for frame 219 of the ‘Outdoor’ sequence with mean-shift based methods.

information is due to the smaller size of the disparity map, a result of the stereo algorithm used to compute the depth data. (Note that the previous marker guided segmentation was not handicapped by this condition.) The block effect in the motion information is also quite visible in the segmentation. The holes in these markers are also leading to a local over-segmentation of the segmentation. We conclude that this approach can not tolerate such severe degradation in the quality of the auxiliary metadata.

7.4.2 The ‘Indoor’ sequence

Because the comparative study for the ‘Indoor’ sequence followed the same reasoning as for the ‘Outdoor’ sequence, we restrict to present here the attained results in figures 7.18, 7.19, 7.20, 7.21 and 7.22, and table 7.2, together with some points to be noticed.



(a) Left frame 112.



(b) Right frame 112.

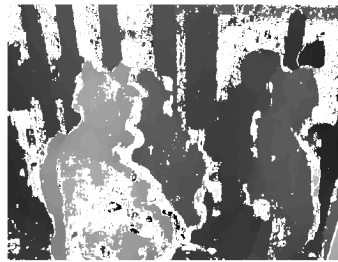


(c) Left frame 113.

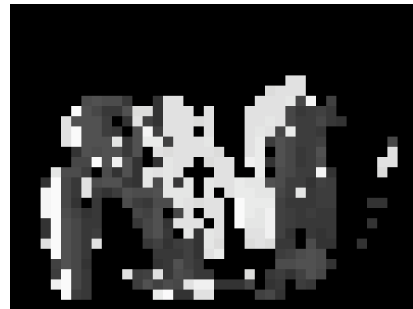


(d) Right frame 113.

Figure 7.18: ‘Indoor’ stereo sequence.



(a) Right frame 113 depth image.



(b) Right frame 113 motion image.

Figure 7.19: Computed depth and motion images for right frame 113 of ‘Indoor’ sequence.

This experiment shows the strengths of the system presented here. The combination of motion and depth information in the marker extraction step leads to a more reliable and consistent segmentation — observe the incapability of separating the men from each other

and from the background when using only depth to guide the watershed algorithm. Motion information improves segmentation results, without assuming motion continuity. In this aspect, the system is general and performs well.

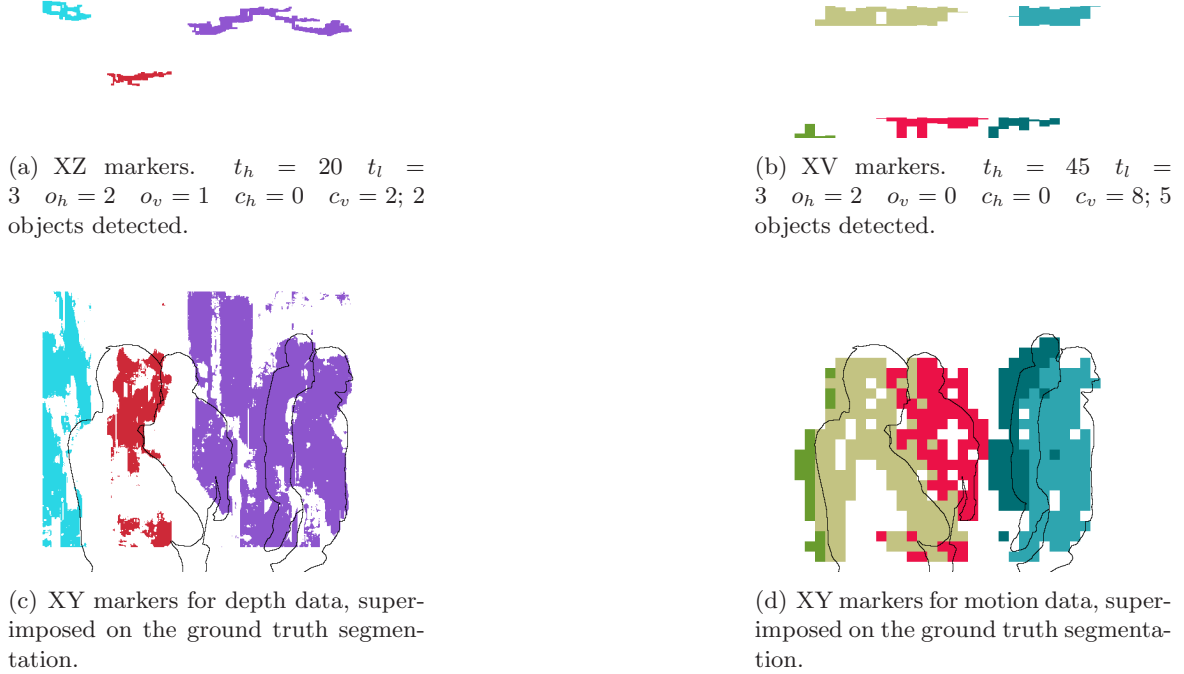


Figure 7.20: Extracted markers for left frame 113 of ‘Indoor’ sequence.

	D	M	DM	mean shift C	mean shift CD	mean shift CM	mean shift CDM
regions	4	5	8	13	38	30	44
d_{sym}	54.52	18.25	28.26	32.17	67.24	53.49	70.67
d_{mut}	18.95	8.67	5.62	15.96	17.99	14.4	14.78

Table 7.2: Results for frame 113 of the ‘Indoor’ sequence. D – depth assisted; M – motion assisted; DM – depth and motion assisted.

7.4.3 Image sequence processing

We completed our study by segmenting a set of 12 consecutive frames of the ‘Indoor’ sequence, from frame 102 to frame 113, image 7.23. All parameters of the different algorithms were kept constant for the whole set. Results are summarized in tables 7.3 and 7.4.

It is visible that the marker based algorithms produce less over-segmented results (smaller number of regions and inferior values of d_{sym}), while maintaining the consistency of the segmentation (d_{mut} value).

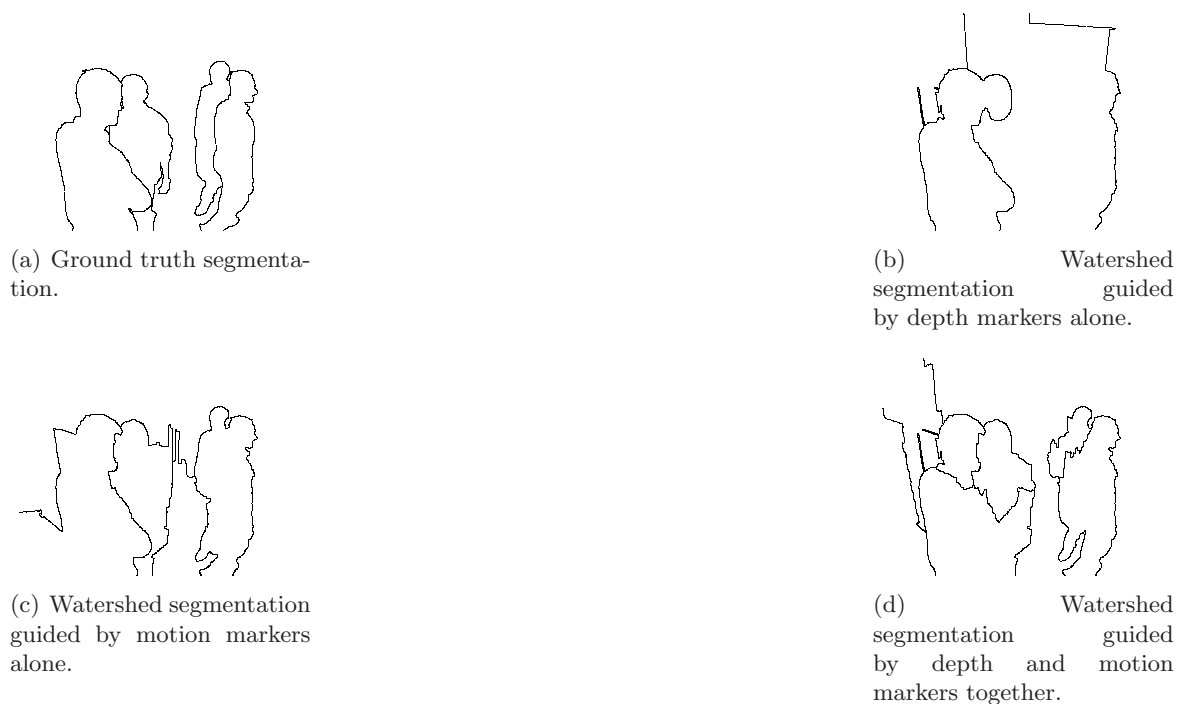


Figure 7.21: Results for frame 113 of the ‘Indoor’ sequence.

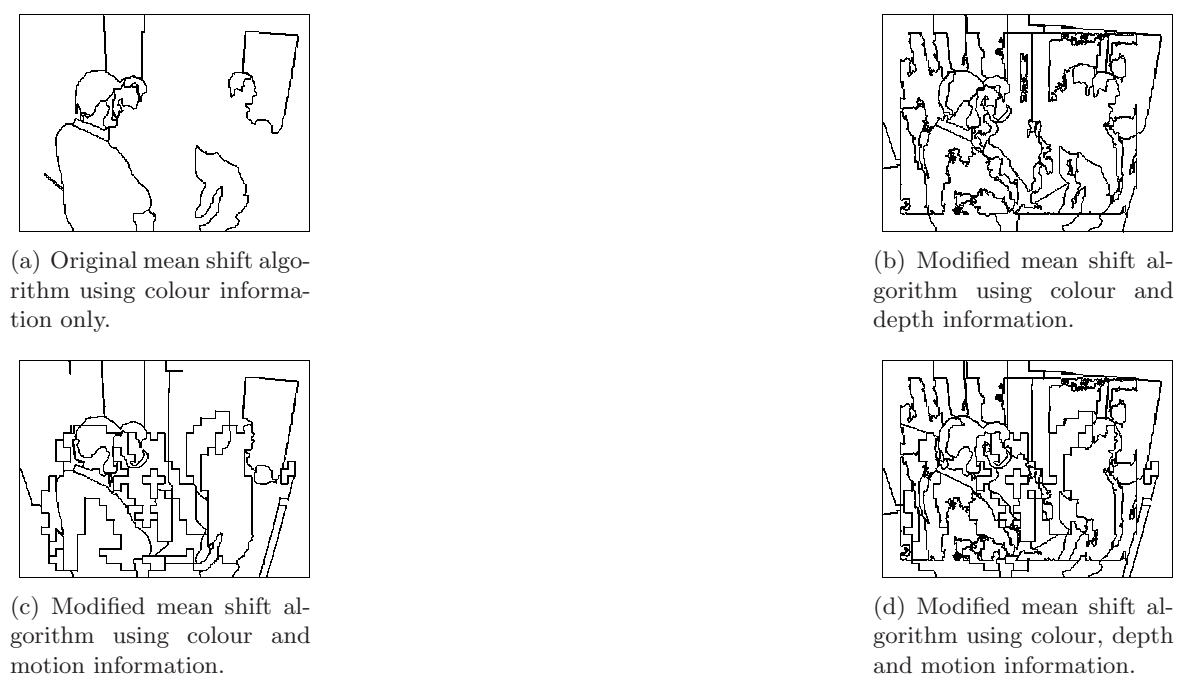


Figure 7.22: Results for frame 113 of the ‘Indoor’ sequence with mean-shift based methods.



Figure 7.23: Twelve frames from the ‘Indoor’ sequence.

Frame	marker D			marker M			marker MD		
	regions	d_{sym}	d_{mut}	regions	d_{sym}	d_{mut}	regions	d_{sym}	d_{mut}
102	4	44.09	15.17	5	18.38	13.55	8	52.36	18.27
103	5	41.07	20.64	5	25.81	15.73	8	37.48	15.08
104	4	44.16	13.91	3	16.24	7.85	6	50.6	14.57
105	4	45.76	12.76	3	23.84	13.63	7	48.73	12.89
106	3	47.11	9.76	4	17.7	12.24	6	39.38	11.88
107	3	42.57	10.95	4	21.47	11.51	6	32.88	11.22
108	3	53.5	20.98	4	16.09	13.62	6	53.73	17.46
109	3	52.21	24.58	6	22.19	15.68	7	35.32	16
110	4	48.95	31.21	5	20.35	19.61	7	49.31	22.61
111	3	39.42	28.23	5	20.02	19.4	7	42.52	25.01
112	4	43.07	32.18	5	19.18	17.74	7	47.28	21.71
mean	3.67	46.37	19.94	4.5	19.96	14.10	6.9	44.51	16.97

Table 7.3: Results for frames 102–113 of the ‘Indoor’ sequence, for marker based methods. D – depth assisted; M – motion assisted; DM – depth and motion assisted.

Frame	mean shift			mean shift D			mean shift M			mean shift DM		
	regions	d_{sym}	d_{mut}	regions	d_{sym}	d_{mut}	regions	d_{sym}	d_{mut}	regions	d_{sym}	d_{mut}
102	9	30.59	13.4	23	58.51	19.1	22	35.08	16.07	33	69.86	18.76
103	10	27.73	9.96	24	65.79	15.15	22	53.93	14.18	37	70.73	14.43
104	9	26.48	9.25	22	66.52	12.67	19	38.09	12.77	31	70.73	11.77
105	10	29.95	15.62	25	70.92	18.28	23	45.18	13.58	33	71.54	10.46
106	11	33.81	16.95	23	69.25	10.16	20	33.84	9.19	34	66.48	9.84
107	10	34.5	14.91	27	64.85	7.97	21	47.43	10.08	33	68.46	9.49
108	12	35.57	17.68	29	61.85	12.88	24	44.37	10.74	38	64.71	8.8
109	11	40.08	21.83	26	55.1	14.13	28	50.33	13.42	36	69.25	17.77
110	16	49.93	13.59	32	62.35	12.09	24	57.92	15.06	42	69.01	15.42
111	13	42.59	20.44	38	59.03	10.33	36	61.82	13.05	41	68.23	15.3
112	18	50.95	17.96	40	62.79	10.66	33	60.91	13.37	47	71.95	16.53
113	13	32.17	15.96	38	67.24	27.99	30	53.49	14.4	44	70.67	14.78
mean	11.83	36.20	15.63	28.92	63.68	14.28	25.12	48.53	12.99	37.42	69.30	13.61

Table 7.4: Results for frames 102–113 of the ‘Indoor’ sequence, for mean-shift based methods. D – depth assisted; M – motion assisted; DM – depth and motion assisted.

7.5 Discussion

The system presented here differs significantly from the established techniques for segmentation from motion and depth. However, most of the components used in this system are techniques known in the literature. One strength of this system is that it performs satisfactorily under severe conditions of noisy in the auxiliary metadata. This was demonstrated by using the output of a simple block motion estimation as the source of motion data, with its block effect (the block size was 16×16) and spatial instability. The flexibility of this system to integrate additional metadata should also not be underestimated. An additional strength is its simplicity, making it suitable for real-time applications.

The type of segmentation performed by the proposed system should be distinguished from those obtained with systems using a sequence of frame with memory instead of a simple pair of consecutive frames. Because no motion continuity is assumed, this system is more general and copes transparently with camera motion, video shot transitions or illumination changes; on the other hand, it expectedly performs worst when motion continuity is verified. The proposed segmentation technique could in fact be used as a building block of a complete tracking system or memory-based segmentation system.

Chapter 8

Conclusion

This thesis focuses on the study of image segmentation techniques assisted by metadata. In particular, it was studied the use of depth and motion information to assist the segmentation process. Several novel approaches to fuse colour and depth information for image segmentation were gauged. A first idea of allowing the use of standard segmentation methods by creating a new image containing information from all sources of data was discarded due to the lack of improvement, comparatively to the use of colour alone. This approach was assessed with two different fusion approaches, obtaining the fused image as a weighted sum of the input images or performing the fusion process with the coefficients of a multiscale transform. Next, we conducted the study by joint-modelling colour and depth information. Adapting a well-established algorithm, we were able to improve the quality of the segmentations. However, this technique also revealed some insufficiencies with images obtained in real-settings, where the depth information is typically noisier than colour information. That led us to further extend this technique with a border refinement step, with positive results.

A last scheme to assist a colour image segmentation with depth information was proposed as a two-step operation: the depth information is initially used to produce object markers, providing a crude identification of the objects in the image; next, a guided image segmentation, starting from the markers, is conducted to refine the regions. This framework constitutes a powerful tool to incorporate information with low reliability in the segmentation process, without being distracted with the noise present in the data. A restrictive assumption of this approach is that objects of interest have large vertical sections. These are necessary to create high-density areas in the density-image.

The study proceeded with the extension of the most promising fusion techniques to incorporate motion information. Using synthetic images we validated the proposed tools in a more generic setting, making simultaneous use of colour, depth and motion data. Then the segmentation techniques were gauged under more difficult conditions. Using real image sequences, depth information was obtained using a stereo algorithm (because the adopted stereo algorithm was suitable for real-time applications, the outputted depth data was rather noisy) and motion with a basic block motion estimation. Under these stressful conditions, the technique based on the mean-shift algorithm started to break, yielding unpleasant segmentations with visible artifacts. On the other hand, the marker based segmentation continued to perform adequately.

Because a fair judgment of any new image segmentation algorithm needs a fair comparison metric, a preliminary study on metrics for comparing image segmentations was conducted in first place. The disappointment with the existing measures led to an exhaustive investigation on new solutions, culminating on the rediscover of the partition-distance measures, introducing them on the image engineering community for the first time. In the numerous reported experiments, it is provided experimental evidence of the adequacy of these measures. It is also worth to stress that, besides providing a value for the overall quality of the segmentation, these measures also offer, for the first time, an image error mask identifying the spatial localization of the errors, a key feature for some applications. It is expected that the partition-distances introduced in this work will become routinely used by most researchers as an indicator of the quality of a segmentation algorithm performance, as they provide a major leap to previous work.

Benefits of the research will accrue from applying the results to coding techniques based on object segmentation, exploiting the availability of additional depth information, beyond the current state of the art. Improved picture quality, affordable 3D content creation and delivery through MPEG-4 SNHC (Synthetic and Natural Hybrid Coding), increased capability to bring more content to the consumer, more artistic freedom and lower costs are the expectations for the enhanced operations with synchronous colour and depth data.

Future work

The studies of this thesis, although with some conclusive results, constitute the starting point for a possible larger project, with focus on multidimensional image modelling and processing, continuing to investigate new ways to fuse data for better image segmentation, making easier important subsequent operations. A first line of evolution concerns naturally the continuity of the research carried out here, improving the performance of the proposed procedures (active contours seem the next natural choice for the guided image segmentation phase after the marker extraction, as it should be robust against markers extending beyond objects' borders) or studying the suitability of other image segmentation frameworks.

Some of the most recent image segmentations techniques approach the problem as a graph partitioning problem, using spectral methods to efficiently attain the solution [25]. Among the most promising techniques we can also find those based on parametric bayesian formulations, imposing spatial coherence by a Markov random field prior or, more recently, with a multinomial logistic regression model that expands the number of possible priors [105]. Dirichlet process mixture (DPM) models have been studied in nonparametric bayesian statistics for more than two decades. Originally introduced by [106] and [107], interest in these models have been applied in statistics to problems such as regression, density, density estimation, contingency tables or survival analysis. More recently, they have been introduced in machine learning and language processing [108]. A first study, yet unsophisticated, on the application of the model DPM to the image segmentation problem, appears in [109]. This approach enables to integrate naturally the evaluation of the number of regions of the image. The best strategy to impose the spatial coherence in the segmentation process still needs to be investigated. Simultaneously, the adoption of a structured representation for the output domain, the segmentation, should boost further the quality of the algorithms. These techniques could all be investigated, as well as their generalization for hybrid images, with information of colour and depth.

A second line of investigation concerns with the application of the partition-distance measures in specific scenarios. In the problem of classifying regions in remotely sensed images, if the weight the edge of the graph if not the intersection of two regions but the intersection scaled by the cost of classifying region A as region B, then we obtain a more sensible measure. An open question here is also the complexity of the mutual partition-distance. Although the partition-distance can be efficiently computed as traditional assignment problem, no such efficient algorithm was found for the mutual partition-distance. The computational complexity of this measure is still an open question.

References

- [1] G. A. Thomas, M. Koppetz, and O. Grau, “New methods of image capture to support advanced post-production,” in *Proceedings of Int. Broadcasting Convention (IBC 2003)*, sep 2003.
- [2] “Metavision ist-99-20 859 european project,” <http://www.ist-metavision.com/>, 2001.
- [3] O. Grau, S. Minelly, and G. A. Thomas, “Applications of depth metadata,” in *Proceedings of International Broadcasting Convention (IBC 2001)*, sep 2001, pp. 62–70.
- [4] G. A. Thomas and O. Grau, “3d image sequence acquisition for tv and film production,” in *Proceedings of 1st International Symposium on 3D Data Processing, Visualisation and Transmission (3DPVT)*, jun 2002.
- [5] 3DV Systems DMC 100 depth machine camera. [Online]. Available: <http://www.3dvsystems.com/>
- [6] The canestavision electronic perception development kit (ep devkit). [Online]. Available: <http://www.canesta.com/>
- [7] Swissranger 3d camera. [Online]. Available: <http://www.swissranger.ch/>
- [8] Pmd[vision] camera-sets. [Online]. Available: <http://www.pmdtec.com/>
- [9] P. W. Walland, G. Thomas, M. Koppetz, J. S. Cardoso, T. Erseghe, and F. Hericourt, “The application of intimate metadata in post production,” in *Proceedings of Int. Broadcasting Convention (IBC 2002)*, sep 2002.
- [10] E. Steinback, P. Eisert, and B. Girod, “Motion-based analysis and segmentation of image sequences using 3-d scene models,” *Signal Processing*, vol. 66, pp. 233–247, 1998.
- [11] G. Dong and M. Xie, “Color clustering and learning for image segmentation based on neural networks,” *IEEE Transactions on Neural Networks*, vol. 16, pp. 925–936, july 2005.

-
- [12] S. Makrogiannis, G. Economou, and S. Fotopoulos, "A region dissimilarity relation that combines feature-space and spatial information for color image segmentations," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 35, pp. 44–53, february 2005.
- [13] J. Liu and Y.-H. Yang, "Multiresolution color image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, pp. 689–700, july 1994.
- [14] S. Ji and H.-W. Park, "Image segmentation of color image based on region coherency," in *Proceedings International Conference on Image Processing*, 1998, pp. 80–83.
- [15] S. C. Zhu and A. Yuille, "Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 884–900, september 1996.
- [16] Y. Deng and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 800–810, august 2001.
- [17] P. Trahanias and A. N. Venetsanopoulos, "Vector order statistics operators as color edge detectors," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 26, pp. 135–143, february 1996.
- [18] M. A. Ruzon and C. Tomasi, "Color edge detection with the compass operator," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 1999, pp. 160–166.
- [19] D. Androutsos, K. N. Plataniotis, and A. N. Venetsanopoulos, "Distance measures for color image retrieval," in *Proceedings International Conference on Image Processing*, vol. 2, 1998, pp. 770–774.
- [20] L. Shafarenko, H. Petrou, and J. Kittler, "Histogram-based segmentation in a perceptually uniform color space," *IEEE Transactions on Image Processing*, vol. 7, pp. 1354–1358, september 1998.
- [21] H. G. Wilson, B. Boots, and A. A. Millward, "A comparison of hierarchical and partitional clustering techniques for multispectral image classification," in *IEEE International Geoscience and Remote Sensing Symposium*, vol. 3, 2002, pp. 1624–1626.
- [22] R. H. Turi, "Clustering-based colour image segmentation," Ph.D. dissertation, Monash University, 2001. [Online]. Available: <http://www.csse.monash.edu.au/~roset/publications.html>
- [23] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603–619, may 2002.

- [24] O. J. Morris, J. Lee, and A. G. Constantinides, "Graph theory for image analysis: An approach based on the shortest spanning tree," *Proceedings IEEE, Part F, Communications Radar Signal Processing*, vol. 133, pp. 146–152, 1986.
- [25] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 888–905, august 2000.
- [26] L. Grady and E. L. Schwartz, "Isoperimetric graph partitioning for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 469–475, march 2006.
- [27] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1101–1113, 1993.
- [28] P. W. Power and R. S. Clist, "Comparison of supervised learning techniques applied to color segmentation of fruit images," in *Proceedings SPIE, Intelligent Robots and Computer Vision XV: Algorithms, Techniques, Active Vision, and Materials Handling*, D. P. Casasent, Ed., vol. 2904, november 1996, pp. 370–381.
- [29] Y. Qi, A. Hauptmann, and T. Liu, "Supervised classification for video shot segmentation," in *Proceedings International Conference on Multimedia and Expo*, vol. 2, july 2003, pp. 689–692.
- [30] N. Vandenbroucke, L. Macaire, and J.-G. G. Postaire, "Color image segmentation by supervised pixel classification in a color texture feature space. application to soccer image segmentation," in *Proceedings International Conference on Pattern Recognition*, vol. 3, 2000, pp. 621–624.
- [31] A. W. M. Kass and D. Terzopoulos, "Snakes: active contour models," *International Journal Computer Vision*, vol. 1, pp. 321–331, 1987.
- [32] J. L. P. Chenyang Xu, "Snakes, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, pp. 359–369, 1998.
- [33] B. Sumengen and B. S. Manjunath, "Edgeflow-driven variational image segmentation: Theory and performance evaluation," UC Santa Barbara, Tech. Rep., May 2005. [Online]. Available: <http://vision.ece.ucsb.edu/publications/05TechRepBaris.pdf>
- [34] G. de Haan, P. Biezen, H. Huijgen, and O. Ojo, "True-motion estimation with 3-d recursive search block matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, pp. 368–379, 1993.
- [35] D. Martin, "An empirical approach to grouping and segmentation," Ph.D. dissertation, UC Berkeley, 2003.

-
- [36] (2002) Berkeley segmentation dataset. [Online]. Available: <http://www.cs.berkeley.edu/projects/vision/bsds>
- [37] Y. Cao, D. Li, W. Tavanapong, J. Oh, J. Wong, and P. C. de Groen, "Parsing and browsing tools for colonoscopy videos," in *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*. New York, NY, USA: ACM Press, 2004, pp. 844–851.
- [38] M. E. Celebi, Y. A. Aslandogan, and P. R. Bergstresser, "Unsupervised border detection of skin lesion images," in *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC 2005)*, vol. 02, 2005, pp. 123–128.
- [39] M. Servais, T. Vlachos, and T. Davies, "Affine motion compensation using a content-based mesh," in *IEE Proceedings on Vision, Image and Signal Processing*, vol. 152, 2005, pp. 31–39.
- [40] Y. Wang, J. Yang, and Y. Zhou, "Unsupervised color-texture segmentation," in *Lecture Notes in Computer Science: Proceedings of the International Conference on Image Analysis and Recognition ICIAR 2004*, vol. 3211. Springer-Verlag, 2004, pp. 106–113.
- [41] K. Fukunaga and L. D. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, pp. 32–40, january 1975.
- [42] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 790–799, august 1995.
- [43] Q. Luo and T. M. Khoshgoftaar, "Efficient image segmentation by mean shift clustering and mdl-guided region merging," in *16th IEEE International Conference on Tools with Artificial Intelligence*, 2004, pp. 337–343.
- [44] O. Debeir, P. V. Ham, R. Kiss, and C. Decaestecker, "Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes," *IEEE Transactions on Medical Imaging*, vol. 24, pp. 697–711, june 2005.
- [45] J. Carballido-Gamio, S. J. Belongie, and S. Majumdar, "Normalized cuts in 3-d for spinal mri segmentation," *IEEE Transactions on Medical Imaging*, vol. 23, pp. 36–44, january 2004.
- [46] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, "Video summarization and scene detection by graph modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, pp. 296–305, february 2005.

- [47] J. S. Cardoso and L. Corte-Real, "Toward a generic evaluation of image segmentation," *IEEE Transactions on Image Processing*, vol. 14, pp. 1773–1782, november 2005.
- [48] —, "A measure for mutual refinements of image segmentations," *IEEE Transactions on Image Processing*, 2006.
- [49] —, "Image segmentation guided by depth information," in *submitted to IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2006 (CVPR2006)*, 2006.
- [50] Hadamard, "Sur les problemes aux derivees partielles et leur signification physique," *Princeton University Bulletin*, pp. 49–52, 1902.
- [51] Y. J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognition*, vol. 29, no. 8, pp. 1335–1346, 1996.
- [52] J. Weszka and A. Rosenfeld, "Threshold evaluation techniques," *IEEE Transactions on System, Man And Cybernetics*, vol. 8, pp. 622–629, 1978.
- [53] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, pp. 1277–1294, 1993.
- [54] M. D. Levine and A. M. Nazif, "An experimental rule-based system for testing low level segmentation strategies," *Multicomputers and Image Processing Algorithms and Programs, Academic Press*, pp. 149–160, 1982.
- [55] M. Borsotti, P. Campadelli, and R. Schettini, "Quantitative evaluation of color image segmentation results," *Pattern Recognition Letters*, vol. 19, no. 8, pp. 741–747, 1998.
- [56] M. D. Levine and A. Nazif, "Dynamic measurement of computer generated image segmentation," in *IEEE Transactions of Pattern Analysis and Machine Intelligence*, vol. 7, 1985, pp. 155–164.
- [57] C. Rosenberger and K. Chehdi, "Genetic fusion: application to multi-components image segmentation," in *Proceedings IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, vol. 4, 2000, pp. 2219–2222.
- [58] P. K. Sahoo, S. Soltani, and A. K. C. Wang, "A survey of thresholding techniques," *Computer Vision, Graphics, and Image Processing*, vol. 41, pp. 233–260, 1988.
- [59] W. A. Yasnoff, J. K. Mui, and J. W. Bacus, "Error measures for scene segmentation," *Pattern Recognition*, vol. 9, pp. 217–231, 1977.
- [60] E. Abdou and W. Pratt, "Quantitative design and evaluation of enhancement/thresholding edge detectors," *Proceedings of IEEE*, vol. 67, pp. 753–763, 1979.

-
- [61] R. Ramón-Roldán, J. F. Gómez-Lopera, C. Atae-Allah, J. Martínez-Aroza, and P. L. Luque-Escamilla, “Measure of quality for evaluating methods of segmentation and edge-detection,” *Pattern Recognition*, vol. 34, pp. 969–980, 2001.
 - [62] Y. J. Zhang and J. J. Gerbrands, “Segmentation evaluation using ultimate measurement accuracy,” in *Proceedings CVPR*, vol. 1657, 1992, pp. 449–460.
 - [63] —, “Objective and quantitative segmentation evaluation and comparison,” *Signal Processing*, vol. 39, no. 1-2, pp. 43–54, 1994.
 - [64] M. F. Mattana, J. Facon, and A. S. Britto, “Evaluation by recognition of thresholding-based segmentation techniques on brazilian bankchecks,” in *Proceedings SPIE*, vol. 3572, 1999, pp. 344–348.
 - [65] Z. M. Huo and M. L. Giger, “Evaluation of a computer segmentation method based on performances of an automated classification method,” in *Proceedings SPIE*, vol. 3981, 2000, pp. 16–21.
 - [66] V. Chalana and Y. Kim, “A methodology for evaluation of boundary detection algorithms on medical images,” *IEEE Transactions Medical Imaging*, vol. 16, no. 5, pp. 642–652, 1997.
 - [67] A. A. Betanzos, B. A. Varela, and A. C. Martínez, “Analysis and evaluation of hard and fuzzy clustering segmentation techniques in burned patient images,” *Image and Vision Computing*, vol. 18, no. 13, pp. 1045–1054, 2000.
 - [68] A. Hoover, G. Jean-Baptiste, X. Y. Jiang, P. J. Flynn, H. Bunke, D. B. Goldgof, K. W. Bowyer, D. W. Eggert, A. W. Fitzgibbon, and R. B. Fisher, “An experimental comparison of range image segmentation algorithms,” *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 18, no. 7, pp. 673–689, 1996.
 - [69] J. Min, M. Powell, and K. W. Bowyer, “Automated performance evaluation of range image segmentation algorithms,” in *Workshop on the Application of Computer Vision (WACV2000)*, California, 2000.
 - [70] K. I. Chang, “Evaluation of texture segmentation algorithms,” in *Proceedings CVPR*, vol. 1, 1999, pp. 294–299.
 - [71] B. Belaroussi, C. Odet, and H. Benoit-Cattin, “Scalable discrepancy measures for segmentation evaluation,” in *Proceedings of IEEE International Conference on Image Processing (ICIP-02)*, 2002.
 - [72] A. B. Goumeidane, M. Khamadje, B. Belaroussi, H. Benoit-Cattin, and C. Odet, “New discrepancy measures for segmentation evaluation,” in *Proceedings IEEE International Conference on Image Processing (ICIP-03)*, Barcelona, Spain, 2003.

- [73] M. Everingham, H. Muller, and B. T. Thomas, "Evaluating image segmentation algorithms using the pareto front," in *Proceedings of the 7th European Conference on Computer Vision (ECCV2002)*, May 2002, pp. IV:34–48.
- [74] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings of the 8th International Conference on Computer Vision (ICCV-01)*, vol. 2, July 2001, pp. 416–423.
- [75] L. Guigues, "Comparison of image segmentations using a hierchical model for n-m regions matching," in *Proceedings of the 2nd IAPR TC-15 Workshop on Graph-based Representations in Pattern Recognition*, Austria, 1999, pp. 41–50.
- [76] A. Almudevar and C. Field, "Estimation of single generation sibling relationships based on dna markers," *Journal Agricultural, Biological and environment statistics*, vol. 4, pp. 136–165, 1999.
- [77] D. Gusfield, "Partition distance: a problem and class of perfect graphs arising in clustering," *Information Processing Letters*, vol. 82, pp. 159–164, May 2002.
- [78] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical recipes in C: the art of scientific computing*. Cambridge University Press, 1999.
- [79] R. Horst, P. Pardalos, and N. Thoai, *Introduction to Global Optimization*. Kluwer Academic Publishers, 1995.
- [80] H. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.
- [81] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. Wiley Interscience, 2001.
- [82] V. Vapnik, *Statistical learning theory*. John Wiley, 1998.
- [83] B. V. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications," *Proceedings of the IEEE*, vol. 85, no. 1, jan 1997.
- [84] —, "Industrial applications of multi-sensor multi-source information fusion," in *Proceedings of IEEE International Conference on Industrial Technology 2000*, vol. 2, 2000.
- [85] L. Wald, *Data Fusion: Definitions and architectures*. Les Presses de l'École des Mines, 2002.
- [86] P. K. Varshney, "Multisensor data fusion," *Electron. Commun. Eng. J.*, pp. 245–253, 1997.

-
- [87] B. Ma, "Parametric and nonparametric approaches for multisensor data fusion," Ph.D. dissertation, University of Michigan, 2001.
- [88] L. G. Brown, "A survey of image registration techniques," *ACM Computing Surveys*, vol. 24, no. 4, pp. 325–376, 1992.
- [89] B. Zitova and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977–1000, october 2003.
- [90] J. Morovic, J. Shaw, and P.-L. Sun, "A fast, non-iterative and exact histogram matching algorithm," *Pattern Recognition Letters*, vol. 23, pp. 127–135, january 2002.
- [91] Z. Zhang and R. S. Blum, "A categorization of multiscale-decomposition-based image fusion schemes with a performance study for a digital camera application," *Proceedings of the IEEE*, vol. 87, pp. 1315–1326, august 1999.
- [92] Vincent, Luc, and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Transactions of Pattern Analysis and Machine Intelligence*, vol. 13, pp. 583–598, 1991.
- [93] P. Soille, *Morphological image analysis*. Springer-Verlag, 1999.
- [94] N. H. Kim and J. S. Park, "Segmentation of object regions using depth information," in *Proceedings of the IEEE International Conference on Image Processing ICIP 2004*, 2004, pp. 231–234.
- [95] Y. Huang, S. Fu, and C. Thompson, "Stereovision-based object segmentation for automotive applications," *EURASIP Journal on Applied Signal Processing*, pp. 2322–2329, 2005.
- [96] F. Tsalakanidou, S. Malassiotis, and M. G. Strintzis, "Face localization and authentication using color and depth images," *IEEE Transactions on Image Processing*, vol. 14, pp. 152–168, 2005.
- [97] M. M. Chang, A. M. Tekalp, and M. I. Sezan, "Simultaneous motion estimation and segmentation," *IEEE Transactions on Image Processing*, pp. 1326–1333, 1997.
- [98] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 780–785, 1997.
- [99] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR1999)*, 1999, pp. 246–252.

- [100] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, “Real-time foreground-background segmentation using codebook model,” (*ELSEVIER*) *Real-Time Imaging*, vol. 11, pp. 172–185, 2005.
- [101] M. Harville, G. G. Gordon, and J. Woodfill, “Foreground segmentation using adaptive mixture models in color and depth,” in *IEEE Workshop on Detection and Recognition of Events in Video*, 2001, pp. 3–11.
- [102] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, vol. 8, pp. 338–353, 1965.
- [103] D. DuBois and H. Prade, *Fuzzy Sets and Systems*. Orlando, FL, USA: Academic Press, Inc., 1980.
- [104] L. Di Stefano, M. Marchionni, and S. Mattoccia, “A fast area-based stereo matching algorithm,” *Image and Vision Computing*, vol. 22, no. 12, pp. 983–1005, Oct 2004.
- [105] M. A. T. Figueiredo, “Bayesian image segmentation using wavelet-based priors,” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2005 (CVPR2005)*, 2005.
- [106] T. S. Ferguson, “A bayesian analysis of some nonparametric problems,” *Annals of Statistics*, vol. 1, 1973.
- [107] C. E. Antoniak, “Mistures of dirichlet processes with applications to bayesian nonparametric estimation,” *Annals of Statistics*, vol. 2, pp. 1152–1174, 1974.
- [108] D. M. Blei and M. I. Jordan, “Variational inference for dirichlet process mixtures,” *Bayesian Analysis*, vol. 1, pp. 121–144, 2005.
- [109] P. O. e Joachim M. Buhmann, “Nonparametric bayesian image segmentation,” *ETH Zürich, Technical Report 496*, 2005.