3D/2D Object Recognition from Surface Patterns

Zhimin Shao
V.S.S.P Group
Department of Electronic and Electrical Engineering
University of Surrey
Guildford, Surrey GU2 5XH

September 1997
To my parents

道可道，非常道。

The Way as it can be given a single interpretation is not The True Way

- Lao Zi (Lao Tzu), 5th Century BC. The Chinese character “道” has many meanings, including “The Way”, “to interpret”, “to tell”. The true meaning can only be determined through the context in which it is placed.
Acknowledgement

I am indebted to my supervisor, Prof. Josef Kittler for his support and guidance, particularly during the difficult times of the work. I also would like to thank every member of the VSSP group for the numerous insightful discussions, suggestions and criticisms.
Abstract

Attributed Relational Graph (ARG) is a powerful representation for model based object recognition due to its inherent robustness in handling noisy and incomplete data. In the past few years, the availability of efficient ARG matching algorithms and their theoretical underpinnings have greatly contributed to many successful applications of ARG representation in tackling high level vision problems.

During my past three year investigation into object recognition using ARG representation, we have developed a number of novel theories and techniques in the subject area. Some are image processing techniques which help to segment and generate primitive features for building ARG representation (Chapter 2 and 4). Some are about projective invariance in ARG representations (Chapter 3 and 5). Some are about new ARG matching algorithms (Chapter 6). This thesis serves as a summary document of these theories and techniques. The most important contributions of our work to the domain of computer vision, in my opinion, are in two areas:

Firstly, in the area of projective invariant ARG representation for object recognition. Here, we demonstrated for the first time, a way to systematically derive ARG representation for objects under complex projective transform by exploiting the knowledge of invariance. The methodology developed by us is a sound strategy that generates ARG representations with a number of desirable and provable properties, amongst which, the most important one is the ability to capture global transformation constraint using binary relations only. The approach significantly reduces the heuristic nature of designing relational measurements and paves the way for wider application of ARG representation in 2D and 3D object recognition.

Secondly, in the area of ARG matching. A new mathematical framework for deterministic relaxation algorithms was developed to overcome a number of problems appeared in the existing theories and practices of efficient ARG labelling. A novel labelling algorithm was proposed based on the new theoretical framework. The algorithm has a number of desirable properties compared to existing algorithms. In particular, the resulting algorithm delivers more consistent, faithful-to-observation results in the presence of ambiguities and multiple interpretations compared to other algorithms.
## Contents

1 Introduction

1.1 Representing 3D Objects as Surface Mapped Polyhedra ...................... 2
1.2 Recognition Strategy for Surface Mapped Polyhedra .......................... 4
   1.2.1 Context dependent labelling .............................................. 6
   1.2.2 Transformation Invariant Object Recognition Using Attributed Rela-
        tional Graph Matching ...................................................... 6
1.3 A Summary of Contributive Ideas Presented in the Thesis .................... 9
   1.3.1 Feature Detection ............................................................ 9
   1.3.2 Affine invariant ARG representation of images using arbitrary contour
        and region features .......................................................... 10
   1.3.3 New ARG matching algorithm and its fuzzy set theoretical background 10
   1.3.4 Extracting multiple label messages from correspondences of 2D fea-
        tures to generate global affine transforms and 3D pose recovery ......... 10

2 Extracting Affine Invariant Points : Edge Based Approach .................. 13
   2.1 Computational Theory of Feature Detection .................................... 15
      2.1.1 New Criteria for Feature Measurement Filtering ........................ 15
      2.1.2 Relations with Canny's Criteria ........................................ 17
   2.2 Angular Curvature ............................................................... 19
      2.2.1 Preliminaries .................................................................. 21
      2.2.2 Analysis of the error in calculating angles ............................ 21
      2.2.3 Experimental Validation ................................................ 23
   2.3 Saliency Measurements That Are Not Based on Angles or Curvatures ...... 24
2.3.1 Saliency as Displacement from a Chord .............................................................. 25
2.3.2 Criteria for Choosing Saliency Measurements ............................................... 25

2.4 Smoothing Using an Optimal FIR Filter ............................................................ 28
  2.4.1 The optimal filter for ramp angle changes ........................................................... 29
  2.4.2 Determining filter size $w$ ................................................................................... 30

2.5 Smoothing using Adaptive filters ............................................................................ 32
  2.5.1 Filter Proposed by Saint-Marc et.al ........................................................................... 33
  2.5.2 Our Adaptive Filter ...................................................................................................... 34

2.6 Remarks and Conclusion .......................................................................................... 36

3 Point Feature Based Invariant Measurement: Unary and Binary Forms 44
  3.1 Introduction .................................................................................................................. 44
  3.2 Projective Transforms .................................................................................................... 46
  3.3 Projective Invariant Unary and Binary Measures ........................................................... 49
       3.3.1 A general form of projective invariant unary and binary measurements based on point features ........................................................... 49
       3.3.2 Basis matrices for various projections .................................................................. 52
       3.3.3 Measurement Error and Stability ........................................................................... 57
  3.4 Experiments .................................................................................................................. 60
  3.5 Conclusions .................................................................................................................. 63

4 Colour Model Based Region Segmentation 75
  4.1 Introduction .................................................................................................................. 75
  4.2 Colour features for image segmentation ........................................................................ 77
       4.2.1 Colour coordinate systems ................................................................................ 77
       4.2.2 Combining intensity and chromaticity ................................................................. 80
  4.3 What is a good segmentation? ...................................................................................... 80
       4.3.1 Spatial and feature domain constraints ................................................................. 80
       4.3.2 Constraint satisfaction in conventional segmentation algorithms ................. 82
  4.4 Region segmentation by relaxation operations ........................................................... 83
       4.4.1 Segmentation using KH Probabilistic Relaxation ................................................. 84
       4.4.2 Segmentation using Markov Random Field Model .............................................. 87
7.1.3 Multiple global transforms from a set of Basic Joint Interpretations .. 142
7.2 Pose recovery from a 2D affine transform hypothesis ............................... 143
  7.2.1 Scaled Orthographic Projection and affine transforms ......................... 143
  7.2.2 Recovering the pose from the planar affine transform ......................... 144
7.3 Experiments ................................................................................................. 149
7.4 Conclusion .................................................................................................... 155

8 Summary and future works ........................................................................... 157
  8.1 Summary ..................................................................................................... 157
  8.2 Future work ............................................................................................... 159
    8.2.1 Feature detection using the criteria functions proposed in our Computational Theory of Feature Detection ............................................ 159
    8.2.2 Modelling Haralick's constraints in relaxation based segmentation algorithms ................................................................. 160
    8.2.3 Using multiple interpretations ............................................................. 160
List of Figures

1.1 Polyhedrae objects with surface patterns ................................................................. 2
1.2 Architecture for recognising 3D object from 2D surface patterns. (The numbers indicate the chapters and sections of the thesis in which the related subject is discussed in detail.) ................................................................. 5

2.1 Architecture of a Feature Detector ............................................................................. 16
2.2 Surface $\theta(x, y) = \arctan \left( \frac{2x}{y} \right)$ ................................................................. 22
2.3 Saliency as the distance to a fixed length chord ....................................................... 26
2.4 Comparison of various saliency measurements (the graphs measure the saliency along the contour of the spanner shape in Figure 2.5) ................................. 27
2.5 Pixelisation noise ........................................................................................................... 28
2.6 Comparison of various filters in presence of small quantisation noise .................... 38
2.7 Comparison of various filters in presence of large quantisation noise ...................... 39
2.8 Comparison of various filters in presence of small Gaussian noise ......................... 40
2.9 Comparison of various filters in presence of large Gaussian noise ......................... 41

3.1 Measurement Stability. Sub-figures arranged from top to bottom, left to right: 3.1.1 has basis size 3 pixels, 3.1.2 has basis size 6 pixels, 3.1.3 has basis size 9 pixels, 3.1.4 has basis size 12 pixels, 3.1.5 has basis size 15 pixels, 3.1.6 has basis size 18 pixels, 3.1.7 has basis size 21 pixels, 3.1.8 has basis size 24 pixels, 3.1.9 has basis size 27 pixels. Note that basis size is measured by area-perimeter ratio, and the basis used in the experiment is an equilateral triangle. .......... 64
3.2 Model of "5": reference points are (43, 11), (20, 11), (10, 60) ......................... 65
3.3 Projection of "5": reference points are (71, 45), (41, 63), (95, 117) ..................... 65
3.4 Normalised model "5" in the affine coordinate system defined by its reference points ................................................................. 65
3.5 Normalised projection of "5" in the affine coordinate system defined by its reference points ................................................................. 65
3.6 Model of "F": reference points are (11, 10), (54, 21), (11, 65) ................................................................................................. 66
3.7 Projection of "F": reference points are (24, 102), (72, 50), (104, 125) .............................................................................. 66
3.8 Normalised model "F" in the affine coordinate system defined by its reference points ................................................................. 66
3.9 Normalised projection of "F" in the affine coordinate system defined by its reference points ................................................................. 66
3.10 Model of "e": reference points are (11, 10), (54, 21), (11, 65) ................................................................................................. 66
3.11 Projection of "e": reference points are (24, 102), (72, 50), (104, 125) .............................................................................. 66
3.12 Normalised model "e" in the affine coordinate system defined by its reference points ................................................................. 66
3.13 Normalised projection of "e" in the affine coordinate system defined by its reference points ................................................................. 67
3.14 Planar Projective Transformations ............................................................................................................................................... 67
3.15 Example contour used for computing the invariant .................................................................................................................. 68
3.16 Translation Invariants ............................................................................................................................................................... 68
3.17 Similarity Invariants ....................................................................................................................................................................... 69
3.18 Affine Invariants .............................................................................................................................................................................. 69
3.19 Perspective Invariants .................................................................................................................................................................... 70
3.20 Model of "star" for testing our binary measure: reference points are labelled with \(i = 1\ldots 10\) ...................................................................................... 70
3.21 Projection of "star" for testing our binary measure: reference points correspond to those in the model ........................................................................... 70
3.22 Experiment 1: Matching a hexagon under perspective transform ................................................................................................. 71
4.1 Post-processing of label probabilities by morphology on noisy image ......................................................................................... 93
4.2 Relaxation with isotropic neighbourhood on noisy image: KH-Probabilistic Relaxation ........................................................................... 95
4.3 Segmentation with our relaxation algorithm on noisy image ........................................................................................................ 96
4.4 Segmentation with our relaxation algorithm on clean image

4.5 Using anisotropic gradient inverted filter to regularise noisy image

4.6 Segmentation with our relaxation algorithm on noise filtered image: using anisotropic gradient inverted filter

5.1 The group of four objects in Configuration 1

5.2 The group of four objects in Configuration 2

5.3 Unary measures obtained from Configuration 1 using basis given by 5.24

5.4 Unary measures obtained from Configuration 2 using basis given by 5.24

5.5 The group of four objects in Configuration 3

5.6 The group of four objects in Configuration 4

5.7 Unary measures obtained from Configuration 3 using basis given by 5.24

5.8 Unary measures obtained from Configuration 4 using basis given by 5.24

5.9 The five point polygon used to designate different binary relations. The vertices are (0, 0), (1, 0), (1, 1), (0, 1) and (0, -1).

5.10 An affine projection of the polygon

5.11 The binary relations between the objects in Configuration 1

5.12 The binary relations between the objects in Configuration 2

5.13 The binary relations between the objects in Configuration 3

5.14 The binary relations between the objects in Configuration 4

5.15 A pair of model regions

5.16 An affine projection of the model regions

5.17 An affine projection of model regions with occlusion

5.18 Back-projection of the model object using the recovered affine transform parameters from occluded scene

5.19 An affine projection of model regions with noise

5.20 Back-projection of the model object using the recovered affine transform parameters from noisy scene

6.1 Experiment 1: Similar Parallelograms – labelling with multiple interpretation.

a). line image of an indoor office scene
b). Parallelograms pulled out by our algorithms
6.2 Experiment 2: Road matching. a) A hand-drawn map. b) An aerial image (with the section that needs to be identified marked as black lines). c) Map sections found ................................................................. 133

7.1 Scaled Orthographic projection of point (0,0) (0,1),(1,0) and (1,1) .... 148
7.2 the Model Object ..................................................................................... 150
7.3 Regions in the Model Object that are used to for matching and generation of pose hypothesis. These regions are characterized by their color which is modeled using a simple Gaussian chromaticity distribution. .......................... 150
7.4 Scene 1.1 .................................................................................................... 150
7.5 Regions in Scene 1.1 that are used to for matching and generation of pose hypothesis. These regions are extracted using the color segmentation algorithms described in Chapter 4 .................................................. 150
7.6 Scene 1.2 .................................................................................................... 151
7.7 Regions in Scene 1.2 that are used to for matching and generation of pose hypothesis. .......................................................... 151
7.8 Scene 2.1 .................................................................................................... 151
7.9 Regions in Scene 2.1 that are used to for matching and generation of pose hypothesis. .......................................................... 151
7.10 Scene 2.2 .................................................................................................. 151
7.11 Regions in Scene 2.2 that are used to for matching and generation of pose hypothesis. .......................................................... 151
7.12 Scene 3.1 .................................................................................................. 152
7.13 Regions in Scene 3.1 that are used to for matching and generation of pose hypothesis. .......................................................... 152
7.14 Scene 3.2 .................................................................................................. 152
7.15 Regions in Scene 3.2 that are used to for matching and generation of pose hypothesis. .......................................................... 152
7.16 Affine hypothesis for Scene 1.1 : generated using the matching result of the algorithms described in Chapter 6 ....................... 152
7.17 The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3 ...................................... 152
7.18 Affine hypothesis for Scene 1.2: generated using the matching result of the algorithms described in Chapter 6 ................................................ 153

7.19 The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3 ................................................ 153

7.20 Affine hypothesis for Scene 2.1: generated using the matching result of the algorithms described in Chapter 6 ................................................ 153

7.21 The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3 ................................................ 153

7.22 Affine hypothesis for Scene 2.2: generated using the matching result of the algorithms described in Chapter 6 ................................................ 153

7.23 The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3 ................................................ 153

7.24 Affine hypothesis for Scene 3.1: generated using the matching result of the algorithms described in Chapter 6 ................................................ 154

7.25 The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3 ................................................ 154

7.26 Affine hypothesis for Scene 3.2: generated using the matching result of the algorithms described in Chapter 6 ................................................ 154

7.27 The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3 ................................................ 154
Chapter 1

Introduction

The ultimate goal of many computer vision systems is to extract important features from image data, from which a description, interpretation, or understanding of the scene can be derived automatically by the machine. Image analysis generally involves the study of feature extraction, segmentation and classification techniques [6]. Classification is a decision making process which identifies (labels) the objects in images based on a collection of visual feature observations. In human vision, observations including colour, shape, texture, motion together with other temporal and spatial context all serve as the evidence supporting this decision making process. Such broad integration of information in conjunction with the soft decision making mechanism of the brain make the human vision system a robust object recogniser. In this thesis, I will focus on the problem of 3D (or 2D) object recognition from 2D surface patterns using both shape and colour information. Here the word recognition means not only to identify the object but also to recover the 3D (or 2D) pose of the object(s) with respect to a predefined coordinate system.

As part this main goal, we will investigate the issues relating to the representation of models and scenes in terms of sets of features, extraction techniques for these features and classification and matching methods which utilise our representations. In this thesis, we are dealing with model based vision. Here and in the rest of the thesis, we will implicitly assume that the models of the objects that we want to recognise are available. These models include the descriptions of both static shape and colour features of the objects.
1.1 Representing 3D Objects as Surface Mapped Polyhedrae

The choice of representation (model) for a 3D object will depend on its purpose and use. It is often affected by three factors. The first is the degree of generality required for representing the shape and surface reflectance of various types of objects. A more general representation normally enables the manipulation algorithms based on it to have a wider utility. The second factor is the degree to which the representation should be easy to be analysed and manipulated by human and/or computer. The third factor relates to the compactness of the representation. The more compact the model, the more models we can store and retrieve with a limited resource. Note that compactness does not always facilitate easy manipulation.

Here we choose to use surface mapped polyhedra (polygons) to represent our 3D (2D) objects, where individual surfaces of polyhedra have planar surface markings. This type of approximation applies to many complex 3D objects, and its representation is very compact and easy to manipulate. Among the many types of objects and scenes that can be represented as pattern mapped polyhedra are buildings, landscapes, furniture, cars etc (c.f. Figure 1.1). The fact that it has been extensively used in computer graphics (in which it is often referred to as “texture mapped polygons”) to manipulate and display 3D worlds shows that it has good generality in modelling real objects, especially those articulated man-made objects.
1.1. REPRESENTING 3D OBJECTS AS SURFACE MAPPED POLYHEDRAE

Polygonal surfaces ensure that the patterns being mapped onto them are planar, thereby allowing us to analyse them using 2D projection model, which is significantly simpler than attempting to recover 3D projection parameters directly.

Polyhedral object recognition has been investigated by many researchers since the very early days of machine vision. However the "block world" assumption has long been criticised for its lack of generality. Even very simple unconstrained indoor scenes may contain objects and background that are difficult to be represented as a small set of polyhedra. Despite the simplistic assumption of the blocked world, a robust recovery of the 3D pose of a simple polyhedra remains to be a challenging problem due to the many degrees of freedom existing in the possible transformations and the complexity of 3D projective geometry [8].

In many cases, surface patterns that exist on many objects can cause a dramatic increase in the volume of the data fed into the recognition algorithms and make them slow and prone to error. With regard to the above problems, we have chosen to use texture mapped polygon representation of objects for its generality. Moreover, the surface pattern information can be used by us to aid the recognition of objects instead of making it difficult. The projection of surface patterns can be well modelled by 2D projections which are simple to estimate. Our approach in general is to recognise (or to at least constrain the possible labels of) the surfaces, and then derive the 3D pose information from the correspondences of surfaces. This approach is particularly effective when objects have distinct structural patterns or shapes on their surfaces. Note that this 3D from 2D approach is different from the Characteristic View [3] approach in 3D vision in the sense that we assume the patterns are restricted to 2D surfaces. Our approach is also different from Surface Descriptor method [4] of understanding 3D objects because we assume our surfaces are planar and instead of focusing on the shape of the surface, we use both patterns on the surfaces and the shape of the surface (if available) to determine the its labels.

We assume that the size of the object is small as compared to the object-camera distance. This assumption enables us to introduce two approximations:

- to represent object model with a relatively small set of surfaces without sacrificing a great deal of accuracy, e.g. a block building viewed afar can be assumed to have just four surfaces, even when windows on one side of the building are not exactly co-planar with the wall, the discrepancy as viewed from the camera will be negligibly small.
1.2. RECOGNITION STRATEGY FOR SURFACE MAPPED POLYHEDRA

- to approximate the perspective projections of the surface patterns with affine transform [1].

Note that here we assume that the surface patterns do not extend continuously from one surface patch to the adjacent surfaces, as the region would not be 2D and could not be recovered using a 2D affine projection model. For example, if the surface pattern is a polyline, then the polyline should be a 2D feature and should not extend beyond surface boundary. This assumption of planar feature has limited the type of object we can represent and recognise. It is very difficult if not impossible, to disambiguate projections of 3D features from projections of 2D features from a monocular image without any prior information about the scene. However, since we are dealing with a recognition problem, whether an object contains such ambiguity is known a priori when one constructs its model. This prior information can be used to trigger a context switch in the segmentation process to give a special treatment to those features that are likely to be non-planar. We will show later in the thesis that by carefully segmenting the data using contextual information, and exploiting the shading characteristics of polyhedra, one can often avoid dealing with surface patterns that extend across surfaces.

1.2 Recognition Strategy for Surface Mapped Polyhedra

Our strategy for recognising surface mapped polyhedra is outlined in Figure 1.2. The whole approach is based on recognition of 2D surface patterns. After the surface patterns are identified, surfaces are reconstructed together with surface intersections (3D lines) and 3D vertices. Using the correspondences of vertices and lines, one can reconstruct the 3D pose of the object. In the case that 2D surface patterns are not sufficient to uniquely determine the corresponding 2D surface model, we leave this ambiguity to be resolved by a higher level verification process.

The structure of the thesis also follows this diagram. We will first describe our approaches to the recognition of 2D patterns under affine transform. Colour features and shape features based on points, lines and regions are used to recover the 2D affine transform. Several novel approaches for generating and using 2D features under affine projection have been studied and experimented with. Chapter 2 discusses issues related to the extraction of
1.2. RECOGNITION STRATEGY FOR SURFACE MAPPED POLYHEDRA

Figure 1.2: Architecture for recognising 3D object from 2D surface patterns. (The numbers indicate the chapters and sections of the thesis in which the related subject is discussed in detail.)
2D point features and Chapter 3 describes the affine invariant generated from point features. Chapter 4 addresses the problem of extracting region features using colour and grey-level information. Chapter 5 describes affine recovery from region features. The correspondences of these features in the model and scene are then established using the matching algorithms proposed in Chapter 6. The recovery of 3D information from 2D correspondences will be addressed in Chapter 7.

1.2.1 Context dependent labelling

In high level computer vision, we often need to solve the problem of finding the correspondence between a large number of feature objects. Here the word feature object refers to the smallest entities involved in the matching. They can be individual features and should not be confused with the real world object needed to be recognised by the system. Most of the time, these feature objects are interacting with each other, and the label of one object does not only depend on the unary measurement of the object alone but also the environment where the object is placed, i.e. the labelling is contextually sensitive. The context which the label may depend on includes the labels of the neighbouring objects and the interaction (relational measurement) between the neighbouring object and the object of interest.

In non-contextual labelling, no interactions between the objects is considered. Thus the probability of label taking a particular value is only a function of the unary measurement. As the error in the unary measurement increases, the probability of taking wrong label increases quickly. In contrast, contextual labelling decides the object label by the unary measurements as well as the labels of the neighbouring objects and the relational measurements between them. Thus the effect of the error on a particular measurement on the labelling is reduced, and the labelling becomes more robust. In the case of transformation invariant object recognition, context dependent representation and matching have been found to be quite useful.

1.2.2 Transformation Invariant Object Recognition Using Attributed Relational Graph Matching

Note that when the camera-object distance is much greater than the size of the object, the surfaces patches of the object model and their correspondent regions in the image are related to each other by affine transforms. The invariant recognition of these surface patches under
affine transform is a crucial step towards achieving the recognition of the entire object.

If we are able to separate the objects in the image from the background, the recognition can be done by comparing the invariant representations of the object and the model. Many studies have been reported on transformation invariants [7], and how to use them for recognition purpose [?] [21]. However, in many realistic situations, we can rarely isolate the complete object from the background in the presence of clutter and occlusion. Invariant matching based approaches like affine invariant Fourier descriptors and moment invariants [7] [?] [21] may easily fail when applied to cluttered and incomplete data. Unfortunately this is the case for our surface patch recognition. Specifically, to recognise the surface patches under affine transform, we need firstly to segment the object surface patches from each other and from the background, and then compare the invariant representations of these patches. This is very difficult given the fact that our surfaces may contain many structural features which may conceal and disrupt the true surface boundaries in the image. Indeed, our surface patches may contain such complex projection variant patterns that there are no simple homogeneous features that can be used to characterise them, and hence it is impossible to separate them from the background using any low level segmentation or perceptual grouping process. Note that the as the features of a surface patch under investigation are mixed and concealed among the features of the background and of other objects, segmentation is necessary in order to generate affine invariant representation of the surface patch. If we cannot segment the surface patch, we cannot represent it in terms of global affine invariant measurements such as those described in [7] [21].

One way to get round this problem is to represent both image and the models as collections of local image features and find the partial inexact matching between these two feature sets. Since the features are extracted locally, the damage caused by occlusion is expected to be limited at local level. Partial matching allows extraneous features to be unmatched and thus no prior segmentation or grouping of features is necessary. Inexactness allows image features to deviate from those of the model as a result of noise. Mathematically, the problem of image feature matching has been formulated as one of Attributed Relational Graph (ARG) matching [1] [3]. Local image features such as segments of polygonal or spline approximations of the contour [16] [22] [23], colour clusters [6] and regions [17] are often depicted as the nodes of the ARG. The geometric arrangement of these features representing the structure
of the image is captured by the relational measurements such as distance, angle, adjacency etc. These relational measurements are associated to the arcs of the ARG image representation. Matching, in this context, refers to a process that finds the correspondences between the nodes of the two ARGs, one representing the observed image (the scene), the other representing the model.

A central theme throughout our approach to the recognition problem is to use such distributed representation of model and scene and to use robust contextual dependent labelling methods to match the ARGs. Chapters 2 and 4 are involved with the extraction of local edge and region features respectively. These are areas that have been well studied by many computer vision researchers. In Chapters 2 and 4 we will mainly present the new methods for detecting low level features developed by us, and compare our methods with existing ones.

Previous to our work, Attributed Relational Graph has been successfully used in several high level shape matching problems under a Euclidean transform. The unary and binary measurements used in the graph representations of models and scenes were designed heuristically and were based on specific types of features such as line segments provided by the polygonal approximations of the shape or by the hough transform. Chapter 3 and Chapter 5 develop systematic methods for representing the surface patches under affine transform using Attributed Relational Graph. The importance of our methods is that we show, for the first time, that one can embed the global projective transformation constraint using binary measurements alone, thus enabling us to match the ARGs using efficient deterministic labelling algorithms. Also unlike existing graph matching based approaches, our ARG representations are capable of representing arbitrary image features that are not limited to line segments or points. The resulting binary and unary measurements are orthogonal to each other.

Chapter 6 is dedicated to the problem of contextual dependent matching, with examples of using 2D feature representations described in Chapter 4 and 5 to find optimal affine correspondences. A number of existing algorithms have been studied and compared. A new labelling algorithm based on fuzzy set theory has been proposed, which offers the ability to handle multiple interpretations.
1.3 A Summary of Contributive Ideas Presented in the Thesis

In the thesis, we will present a number of contributive ideas, new theories and novel algorithms that have been developed during our research into the problem of recognition of 3D objects from surface patterns.

1.3.1 Feature Detection

In chapter 2, we present a new theory for feature detection. We argued that criteria for optimal computation of "featured-ness" measurement should be tuned so as to match the characteristics of the thresholding and non-maxima suppression processes. Based on this rationale, we have proposed two criteria for optimal saliency signal processing. The advantages of the new theory are as follows.

1. Our criterion functions provide an objective measurement about how good a signal is for non-maxima suppression and thresholding. The functions are local and easy to measure, and are also independent of the type of processing involved in generating the signal. They can be used directly in the experiments as objective measures to compare the quality of different feature detection algorithms as well as the quality of the raw data. In Section 2.3, we have used these criteria to compare and choose raw saliency measurements.

2. The criteria proposed by us can be directly used to design a wide range of feature detection methods, i.e. they are not just limited to FIR filtering. We have shown how to design adaptive filters that maximise these criteria.

Also we have shown the relationship between our criteria and Canny's criteria, and the relationship between our criteria and adaptive Saint-Marc filter.

Equation 2.19 gives a point dependent formula of angular noise. When the gradient of edges is relatively fixed in a certain region, we can substitute the error estimates with their average, and provide a method of dynamically determining the appropriate filter size based on the specification of the signal and local noise conditions.

In Chapter 4, we present a new method that combines both chrominance and intensity information to achieve model based planar region segmentation in colour images using Relaxation Labelling. We show that our labelling algorithm has justifiably better performance
than many conventional segmentation algorithms. We argue that the better performance is
due to the fact that we have derived our algorithm from the weak-membrane model which
we consider as a good model for segmentation process according to the constraints proposed
by Haralick [3].

1.3.2 Affine invariant ARG representation of images using arbitrary contour and
region features

In Chapter 3 and Chapter 5, we present our affine invariant Attributed Relational Graph
representations of planar shapes. We show for the first time that, by organising the measure-
ments according to our proposed form, it is possible to decompose the global transformation
constraint into a set of unary and binary measures. Measurements based on salient contour
points and/or line segments are presented in Chapter 3, and those based on regions are pre-
sented in Chapter 5. The computational complexity and the robustness of our measurements
are also discussed.

1.3.3 New ARG matching algorithm and its fuzzy set theoretical background

In Chapter 6, we discuss a number existing approaches to the problem of Attributed Rela-
tional Graph Matching and in particular the deterministic relaxation algorithms. We analyse
a number of common problems that exist in the deterministic relaxation algorithms. We
present a new fuzzy set theoretic framework for deterministic relaxation labelling which
enables us to develop relaxation schemes that are free of these problems, and are able to
provide multiple interpretations of a single object, thus achieving many-to-many mapping.
We argue that such many-to-many mappings are important in many vision systems. A
non-iterative attributed relational graph matching algorithm based on the fuzzy theoretical
framework is presented.

1.3.4 Extracting multiple label messages from correspondences of 2D features to
generate global affine transforms and 3D pose recovery

The final stage of recognition is to recover the 3D pose information of the objects in the scene.
As described before, our approach is to obtain the affine hypothesis about the object surfaces
in the scene first, and then derive the 3D pose from 2D affine correspondences.
At the end of the ARG matching process, we have obtained many-to-many correspondences between the model features and scene features, the next step is to retrieve the affine hypothesis from the mapping. The over-determined system of equations involved in “transformation from correspondences” problem can be solved by the Least Square Method. Unfortunately the method only works if the correspondence is one-to-one and good initial guess is available. Traditional methods such as Hypothesis Clustering, Hypothesis-and-Verification are able to deal with many-to-many mappings but also appear to be either reliant on knowledge about the underlying parameter clusters or sensitive to noise. In the first part of the Chapter 7, we show that the fuzzy set theory based on which we developed our matching algorithm also provides a theoretical basis on which we can extract multiple 1-to-1 mappings from the label probabilities as multiple global interpretations (i.e. multiple affine hypothesis.) Our label message extraction method is closely tied with our object centered labelling algorithm such that it is able to reuse the intermediate computation results of the labelling algorithm. This makes our method very fast. The resulting affine transforms from the 1-to-1 correspondences can be optimized using standard least-square based methods.

To date, several formulas of 3D pose recovery from 2D point correspondences have been proposed. In the second part of the chapter, we also give a derivation. The difference between our solution and the previous ones is that our solution for depth and other parameters is given solely in terms of six 2D affine parameters estimated from the planar feature matching results. Instead of requiring point correspondences, our method only requires six affine parameters which can be estimated from correspondences of varieties of features such as regions, texture, curves.
Bibliography


Chapter 2

Extracting Affine Invariant Points: Edge Based Approach

As described by Marr in his famous book about representational paradigm of vision, the analysis and understanding of a scene are always built upon a specific representation of the model and data. To build a representation, one needs first to select a number of features as the basic building blocks of the representation. We take the view that many 3D objects are fully or partially defined in terms of planar patches and our ultimate aim is to recognise such objects based on the shapes of these planar regions and their surface markings under affine transform. This chapter and Chapter 4 address the issue of extracting useful features (from edges and from regions respectively) on which we will build our representation of planar patterns.

Contour curvature features are one of the most important information that can be used for 2D object recognition. The points of maximum convexity, concavity and inflexion partition the image curves into relatively stable segments under 2D projection which then can be matched against each other, or be used to reconstruct other features. We call these points salient points on the curve. The major challenge is to accurately extract these feature points from digitised images in the presence of noise. Two questions need to be answered in order to achieve our goal, one is what type of measurement we should use to capture our notion of saliency, the other is how do we process the chosen measurement to extract feature points reliably.

In this chapter we propose a Computational Theory of Feature Detection inspired by Canny’s Computational Theory of Edge Detection [3]. The theory allows us to express and
answer both questions in a formal fashion. We show that both selection of measurements and processing (filtering) of the obtained measurements should be performed to maximise the objective functions proposed by our Computational Theory of Feature Detection. Based on this argument, we propose a novel saliency measurement and a novel adaptive filter which are justifiably better than conventional approaches.

Section 2 presents the Computational Theory of Feature Detection. We extend the utility of the criterion of SNR and of Localisation to both measurement selection and measurement filtering processes. We show how to experimentally measure these criterion functions. Our theory is not restricted to the analytical forms originally proposed by Canny [3]. It can, for example, serve as a criterion for comparing performance of adaptive and non-linear filters as well.

Sections 3 and 4 discuss the issue of generating the initial measurements and quantifying the noise in the measurements.

In Section 3, we give a quantitative analysis of the noise affecting the angle measurements which is often generated as a by-product of edge detection. The result of the analysis shows that the noise on the angle information is only related to the strength of the edge, and therefore adaptive smoothing according the local noise condition is desirable. The analysis given in Section 3 forms the basis for choosing filter size in Sections 5 and 6 when angle differential (curvature) is used as saliency measurement.

Section 4 describes a number of alternative saliency measurements that one can use when the angle information is not directly available. These include a novel simply computable saliency measure which has been proposed as a good alternative to angular curvature. Sections 5 and 6 develop filters for smoothing the measurements for accurate extraction of point features.

Sections 5 and 6 present the proposed smoothing filters for detecting extrema and zero-crossings on the saliency/curvature measurements. Sections 5 proposes an FIR filter that is optimised according to the proposed criteria for the detection of salient points. We argue that this is a justifiably better FIR filter than the commonly used Gaussian filter. Iterative filters for curvature feature detection are discussed in Section 6 where a new adaptive filter is presented. Experiments show that the proposed iterative filter outperforms median, gaussian and optimal FIR filters in the presence of Gaussian additive noise and quantisation
noise both in terms of Signal to Noise Ratio and Localisation.

Section 7 describes the thresholding and final extraction of points of interest, and demonstrates the result of our feature point detection algorithm.

2.1 Computational Theory of Feature Detection

This section presents a theory of feature detection. By detecting features, here we specifically refer to a two-stage process as illustrated in Figure 2.1. The first stage is the process of computing a "featured-ness" (degree of saliency) measurement in one or two dimensional space. The second stage locates the peaks (i.e. significant local maxima) of this "featured-ness" measurement. Note that the process of locating significant local maxima is relatively well understood. It often consists of Amplitude Thresholding (constant or adaptive) and Non-Maxima Suppression. On the other hand, the first stage of feature detection which involves computing a "featured-ness" measurement from noisy raw data is often found to be challenging. For example, in edge detection problems, this involves designing edge detection masks to filter and compute the gradient from noisy images; in the problem of corner detection, this involves designing filters to give smooth and accurate curvature/saliency estimates. Our Computational Theory of Feature Detection is developed to address the issues in the first stage of feature detection, namely the problem of finding optimal process that computes the "featured-ness" measurement from noisy data.

2.1.1 New Criteria for Feature Measurement Filtering

Consider a 1D "featured-ness" measurement $f(x) = s(x) + n(x)$, where $s(x)$ is the signal component of $f(x)$ and $n(x)$ is the noise component. We propose two criteria which need to be maximised for $f(x)$ to be optimal for thresholding and non-maxima suppression. The first is the SNR Criterion,

$$S_y = \frac{s(x)}{E(n^2)} \approx \frac{f(x)}{E(n^2)}$$

(2.1)

The rationale for this criterion is that since $f(x)$ represents the degree of "featured-ness", the larger this value compared to the noise level, the easier for the peaks to be picked up by
2.1. COMPUTATIONAL THEORY OF FEATURE DETECTION

Figure 2.1: Architecture of a Feature Detector
thresholding. The second is the Localisation Criterion,

\[ L_g = \frac{s''(x)}{E(n^2)} \approx \frac{f''(x)}{E(n^2)} \]  

(2.2)

If we use \( f(x) \) directly for amplitude thresholding and non-maxima suppression, it is desirable that the peaks of the signal are as sharp as possible. Criterion \( L_g \) which measures the curvature of the signal captures this idea. The greater the curvature at the peak point, the sharper the peak. Note that both criteria should be of large value at the anticipated salient points.

### 2.1.2 Relations with Canny's Criteria

In [3] Canny formulated three separate criteria for good extrema detection for 1D signal \( e(x) \) using a fixed filter \( g(x) \), namely the good detection criterion (maximum SNR):

\[ S_c = \frac{\int_{-w}^{w} g(x)e(-x)dx}{E(n^2) \sqrt{\int_{-w}^{w} |g(x)|^2 dx}} \]  

(2.3)

localisation criterion :

\[ L_c = \frac{s^2 \int_{-w}^{w} g'(x)e'(-x)dx}{E(n^2) \sqrt{\int_{-w}^{w} |g'(x)|^2 dx}} \]  

(2.4)

and suppression of false response:

\[ C = \frac{1}{w} \sqrt{\frac{\int_{-w}^{w} |g'(x)|^2 dx}{\int_{-w}^{w} |g''(x)|^2 dx}} \]  

(2.5)

where \( w \) is the filter width and \( s \) is a scaling factor. Note that these criteria are suitable for optimising the detection of an arbitrary feature in \( e(x) \) with a fixed convolution kernel \( g(x) \). In the original Canny's paper [3] and the subsequent developments [4], function \( e(x) \) is assumed to be of step shape. The work was also extended to include ramp shaped signals [13]. We would like to emphasise that the “step-like” shaped signal is quite different from what we call saliency measurements since the \( |e(x)| \) in 2.3 is not a measure of the degree of “featured-ness” in the data. In these edge detection applications, the “featured-ness” measurement is the image gradient \( f(x) = e'(x) \). Since differentiation and filtering can be combined into a single fixed linear convolution kernel \( g(x) \), the differentiation which is a
stage of feature generation was not explicitly separated from the low-pass filtering which is a stage of feature noise removal in Canny’s formulation. In our analysis, we explicitly distinguish the differentiation from the low-pass filtering to map the problem in the form of our feature detection theory. Accordingly we assume that the gradient signal \( f(x) \) is calculated from \( e(x) \) first and then put through a low-pass filter \( h(x) \). This however does not require a modification of Canny’s criteria in 2.3 and 2.4 since they do not make any assumption about the type of the signal and the type of convolution kernel. Therefore we can write the same criteria for optimising the low-pass filter \( h(x) \), given gradient signal \( f(x) \).

The SNR criterion is

\[
S_h = \frac{|\int_{-w}^{w} h(x)f(-x)dx|}{E(\eta^2)\sqrt{\int_{-w}^{w} |h(x)|^2 dx}}
\]  

and the localisation criterion is

\[
L_h = \frac{s^2 |\int_{-w}^{w} h'(x)f'(-x)dx|}{E(\eta^2)\sqrt{\int_{-w}^{w} |h'(x)|^2 dx}}
\]  

If we put saliency measurement \( f(x) \) through a low-pass filtering process, we certainly want the process to try to maximise both the signal value and the signal curvature at the salient points in order to make thresholding and non-maxima suppression effective at picking up the peaks. This is the essence of our SNR and Localisation Criteria in equations 2.1 and 2.2. One can easily see that objective functions \( S_g \) and \( L_g \) in equations 2.1 and 2.2 are consistent with the established criteria \( S_h \) and \( L_h \) in equations 2.6 and 2.7. Canny’s criteria \( S_h \) and \( L_h \) are in fact special cases of our general criteria, when we assume saliency measurement \( f(x) \) is put through a fixed low-pass filter \( h(x) \). One can verify this by letting \( h(x) \) be an all pass filter,

\[
h(x) = \delta(x)
\]  

where \( \delta(x) \) is the Dirac Delta Function, and see what Canny’s criteria \( S_h \) and \( L_h \) are actually optimised for. Note that the convolution of \( f(x) \) with the derivative of the Dirac Delta Function gives \( f'(x) \). Substituting 2.8 into 2.6 and 2.7, we have

\[
S_h = \frac{|\int_{-w}^{w} \delta(x)f(-x)dx|}{E(\eta^2)\sqrt{\int_{-w}^{w} |h(x)|^2 dx}} = \frac{|f(x)|}{E(\eta^2)\sqrt{\int_{-w}^{w} |h(x)|^2 dx}} = \frac{|f(x)|}{E(\eta^2)}
\]
and

\[ L_h = \frac{s^2 \int_{-w}^{w} \delta(x) f'(-x) dx}{E(\eta^2) \sqrt{\int_{-w}^{w} h'(x)^2 dx}} = \frac{s^2 |f''(x)|}{E(\eta^2) \sqrt{\int_{-w}^{w} h'(x)^2 dx}} = \frac{s^2 |f''(x)|}{E(n^2)} \quad (2.10) \]

Equations 2.9 and 2.10 suggest that if we use \( f(x) \) directly for thresholding and non-maxima suppression, Canny's criteria will be a measure of the magnitude and the curvature of the signal relative to noise. This confirms that Canny's criteria do optimise the filter response in the way that is consistent with our proposal.

Note that, unlike Canny's formulation, our criteria \( S_g = \frac{|f(x)|}{E(n^2)} \) and \( L_g = \frac{|f''(x)|}{E(\eta^2)} \) do not attempt to model the underlying filtering process from which the signal estimate of \( f(x) \) is obtained. On the contrary, we focus our attention on the characteristics of the resulting signal and establish our objective functions based on these characteristics. The benefit of our approach is that it can be used to compare and optimise a much wider class processing algorithms including those iterative and/or non-linear filters.

In theory one needs to know the output noise power \( E(n^2) \) in order to use criteria \( S_g \) and \( L_g \) directly to optimise the filtering process. In practice, only in a few special circumstances (as shown in the next Section) can we tell quantitatively about the noise present in the raw signal. However, assuming input noise power \( E(\eta^2) \), it is possible for us to show the relationship between the output noise power \( E(n^2) \) and the input noise power \( E(\eta^2) \) given the details about the filtering process. In case that the signal is processed by a fixed linear filter, the input and output noise power are related by a constant factor \( \sqrt{\sum h(x)^2} \) and \( \sqrt{\sum h'(x)^2} \) as shown in Canny's criteria in 2.3 and 2.4. Usually the noise reduction factor is related to the filtering process, and is used together with \( |f(x)| \) and \( |f''(x)| \) to jointly bare on the optimisation problem.

2.2 Angular Curvature

In conventional mathematics, curvature is defined as the rate of the angle change along the arc length. Although some researchers argued that saliency can be better captured by other means in the digitised image (e.g. computational F/B saliency, see [6], Section 3), curvature remains to be a generic concept for measuring saliency on curves.

Generally there are two classes of approaches to the problem of restoring angle and curvature information from a digitised noisy image. In [19], Worring gave an interesting com-
2.2. ANGULAR CURVATURE

Comparison of the two classes of approaches. Here we summarise Worring's results.

The first class of methods is called orientation based, where the curvature estimation is obtained from angle information. Within this class of methods, there are two approaches: one obtains the angle information from the set of discrete edge pixel positions (e.g. [1],[14]); the other uses angles estimated from other sources such as the gradient information (e.g.[5] [2]). Obviously, if there is no gradient information available, such as in estimating curvature from binary images (which is common in hand-written character recognition), we are constrained to use the first approach. Legault [8] gave a good comparison between the methods that extract curvature from binary images.

The second class of methods is called path based. In these methods, curvatures are calculated indirectly from the parametric approximation of the curve. The parametric model of the curve can either be global or local. Global curve fitting approach includes a number of spline fitting algorithms [12] [10] as well as "snake", and has the drawback of requiring a prior assumption about the form of the underlying contour [16]. Also the problem of optimal piecewise curve fitting (for example, optimal knots insertion in spline approximation of contour [12]) often leads to a minimisation process that is so expensive that people have to settle for suboptimal or heuristic procedures. The other way is to use dynamic local models and treat the evolution of the local model as a filtering process [9] [11] [16]. In particular Sander's Kalman filtering method gives a good performance in terms of accuracy [16]. However such methods require an initial estimate of the appropriate scale of the filter which without knowing the statistical characteristics of the signal and the noise is quite difficult to give (see [16], the example of calculating the curvature of a logarithmic spiral). In general, path based methods perform poorly, giving a large bias on the angle and curvature estimates [19].

In this section, we focus on the orientation based approach, assuming the angle information is directly calculated from the grey level gradient. Our method is different from the previous approaches in that we begin with a quantitative analysis of the noise present in the angle measurement. The result of this analysis will guide our smoothing and detection process adaptively according to the local noise condition.
2.2. ANGULAR CURVATURE

2.2.1 Preliminaries

Image restoration refers to the process of recovering measurements of the objects from noisy and imperfect image data. Image data acquired from a sensing-recording system usually suffers from both image dependent and image independent noise. Our ability of recovering the original signal from noisy data depends on the extent of our knowledge about the degradation process [7](Chapter 8, pp.267). Here we specifically deal with the problem of recovering angle and curvature measurements from grey level images with image independent noise.

Most edge detection processes involve the thresholding of the gradient image. Edge pixels are characterised by local maxima of the gradient values. Let the gradient of image \( f(x,y) \) be:

\[
g(x,y) = \nabla f = \sqrt{g_x^2 + g_y^2}, \quad \text{where} \quad g_x = \frac{\partial f}{\partial x}, g_y = \frac{\partial f}{\partial y}
\]  

(2.11)

The direction of the edge is given by:

\[
\theta(x,y) = \arctan\left(\frac{g_y}{g_x}\right)
\]

(2.12)

In the digital image, \( g_x \) and \( g_y \) are calculated by gradient masks. If the grey-level image is corrupted by an image independent gaussian noise with standard deviation \( \sigma_{image} \), then we can assume that the outputs of the gradient masks are

\[
g_x = g_{x0} + \eta_x, \quad g_y = g_{y0} + \eta_y
\]

(2.13)

where \( g_{x0} \) and \( g_{y0} \) are the true gradients, and \( \eta_x \) and \( \eta_y \) are gaussian variables with covariance

\[
\Sigma = \begin{pmatrix}
\sigma_x^2 & \sigma_{xy} \\
\sigma_{xy} & \sigma_y^2
\end{pmatrix}
\]

(2.14)

Note that \( \Sigma \) is completely defined by the edge detection mask and \( \sigma_{image} \).

2.2.2 Analysis of the error in calculating angles

As Equation 2.12 suggests, the angle measurements can be treated as a two dimensional function of \( x, y \) gradients. This function is discontinuous at \( g_y \)-axis (where \( g_x = 0 \)) where
2.2. ANGULAR CURVATURE

Figure 2.2: Surface \( \theta(x, y) = \arctan\left(\frac{g_x}{g_y}\right) \)

\( \arctan() \) jumps from \( -\pi/2 \) to \( +\pi/2 \). Figure 2.2.2 shows the shape of \( \arctan() \) function.

Notice that, although the surface of \( \theta(x, y) \) is discontinuous, it is continuously differentiable at \(( -\infty, 0) \) and \(( 0, +\infty) \). Also

\[
\lim_{g_x \to 0^-} \frac{\partial f}{\partial g_x} = \lim_{g_x \to 0^+} \frac{\partial f}{\partial g_x} = -\frac{1}{g_y} \tag{2.15}
\]

We can see that partial derivatives \( \frac{\partial f}{\partial g_x} \) and \( \frac{\partial f}{\partial g_y} \) always exist and are finite except at the origin \(( g_x = 0, g_y = 0 )\). Hence we may apply the Taylor expansion to Equation 2.12 after substituting from Equation 2.13:

\[
\begin{align*}
\theta(g_x, g_y) &= \theta(g_{x0} + \eta_x, g_{y0} + \eta_y) \\
&= \theta(g_{x0}, g_{y0}) + (\frac{\partial \theta}{\partial g_x} \eta_x + \frac{\partial \theta}{\partial g_y} \eta_y)|_{(g_{x0}, g_{y0})} \\
&\quad + (\frac{\partial}{\partial g_x} \eta_x + \frac{\partial}{\partial g_y} \eta_y)^2 \theta|_{(g_{x1}, g_{y1})}
\end{align*}
\tag{2.16}
\]
where \( g_{x1} \) and \( g_{y1} \) are bounded by \( g_{x0}, g_{x0} + \eta_x \) and \( g_{y0}, g_{y0} + \eta_y \). Let us approximate the error of the angular measurement \( \eta \) with the 1st order remainder:

\[
\eta(g_{x0}, g_{y0}) = \left( \frac{\partial \theta}{\partial g_x} \eta_x + \frac{\partial \theta}{\partial g_y} \eta_y \right) |_{(g_{x0}, g_{y0})} \tag{2.17}
\]

By differentiating Equation 2.12, we can derive the mean and variance of the error of the angular measurement:

\[
E\{\eta\} = 0 \tag{2.18}
\]

\[
E\{\eta^2\} = \left( \frac{\sigma}{g_0} \right)^2 - 2\frac{g_x g_y \sigma_{xy}}{g_0^4} \tag{2.19}
\]

If the \( x, y \) edge detection masks are separable, then they will not introduce correlations between \( \eta_x \) and \( \eta_y \), and the above equation becomes

\[
E\{\eta^2\} = \left( \frac{\sigma}{g_0} \right)^2 \tag{2.20}
\]

Equation 2.18 and Equation 2.19 have several implications:

1. Equation 2.18 shows that \( \theta(x, y) \) is an unbiased estimate of the angle at point \((x, y)\).

2. The mean square error (which somehow represents the uncertainty) of the angle measurement is only related to the type of edge detection masks used and the gradient measurement (which represents the "strength" of the edge) at the point. Given an estimate of the noise power in an image, we should be able to tell quantitatively about the uncertainty of our angle measurement at any arbitrary point using the given relation.

3. By utilising the uncertainty information, we can produce various kinds of adaptive filtering techniques to "restore" the angle signal from noisy data with a generally higher accuracy than fixed scale filtering techniques. An example of combining an optimal feature detector with this uncertainty information is given in the next section.

### 2.2.3 Experimental Validation

In order to justify Equation 2.18 and Equation 2.19 in the case of practical edge detector, we performed experiments with a separable optimal edge detector and an optimal 2D edge
2.3. SALIENCY MEASUREMENTS THAT ARE NOT BASED ON ANGLES OR CURVATURES

<table>
<thead>
<tr>
<th>Foreground Grey Level</th>
<th>$\frac{\sigma}{\delta \theta}$</th>
<th>M.S. Error</th>
<th>Estimated M.S. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.1400</td>
<td>0.022133</td>
<td>0.0211</td>
</tr>
<tr>
<td>115</td>
<td>0.1088</td>
<td>0.015470</td>
<td>0.0124</td>
</tr>
<tr>
<td>125</td>
<td>0.0900</td>
<td>0.010740</td>
<td>0.0090</td>
</tr>
<tr>
<td>140</td>
<td>0.0786</td>
<td>0.008520</td>
<td>0.0063</td>
</tr>
<tr>
<td>150</td>
<td>0.0700</td>
<td>0.006260</td>
<td>0.0051</td>
</tr>
<tr>
<td>165</td>
<td>0.0615</td>
<td>0.005212</td>
<td>0.0038</td>
</tr>
<tr>
<td>175</td>
<td>0.0560</td>
<td>0.004038</td>
<td>0.0032</td>
</tr>
<tr>
<td>190</td>
<td>0.0505</td>
<td>0.003402</td>
<td>0.0026</td>
</tr>
<tr>
<td>200</td>
<td>0.0470</td>
<td>0.002227</td>
<td>0.0022</td>
</tr>
</tbody>
</table>

Table 2.1: Mean square error of angle measurements in comparison with the theoretical prediction.

detector. A set of stripe patterns with different angles and gradients are used as testing data. Grey level noise with constant variance are added to the image. The mean square errors of angle measurements are shown in Table 2.1. Note that in Table 2.1 the relation between the mean square error and the noise to signal ratio in the gradient measurement generally follows a quadratic curve that passes the origin, which is the same as predicted by Equation 2.19.

2.3 Saliency Measurements That Are Not Based on Angles or Curvatures

Last section introduced angular curvature measurements. An alternative saliency measurement to curvature is often desirable. Curvature as an infinitesimal concept may not be appropriate or easy to be used on digital images [6]. When the angle information is not directly available, e.g. when the edge is obtained from morphological processing of segmented regions, one has to resort to other methods of angle estimation. Regularisation methods that explore the smoothness constraint such as piece-wise curve fitting or kalman filters often yield biased curvature estimates when the scale of the filter or order of the approximation is not chosen correctly [16].

This section introduces some alternative saliency measurements, and uses the criteria proposed in our Computational Theory of Feature Detection to decide which type of saliency measurement to use in the presence of both Gaussian and quantisation noise. We argue that
2.3. SALIENCY MEASUREMENTS THAT ARE NOT BASED ON ANGLES OR CURVATURES

our choice of Central Saliency gives the best signal to noise ratio as well as localisation.

2.3.1 Saliency as Displacement from a Chord

Consider a chord between 2 points $A$ and $B$ on a curve. The distance of the chord to a third point $C$ which lies on the curve segment between $A$ and $B$ gives a good indication of the saliency of point $C$ (Figure 2.3). A large distance to the chord suggests a large bending of the curve segment.

Suppose the curve segment is described by a list of points $\{(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n), \ldots, (x_N, y_N)\}$, and let point $A = (x_{n-k}, y_{n-k})$, $B = (x_{n+k}, y_{n+k})$. The saliency of point $C = (x_n, y_n)$ is measured by:

$$c_n = \frac{1}{k((x_{n+k} - x_{n-k})^2 + (y_{n+k} - y_{n-k})^2)} \begin{vmatrix} x_{n-k} & y_{n-k} & 1 \\ x_n & y_n & 1 \\ x_{n+k} & y_{n+k} & 1 \end{vmatrix}$$ (2.21)

where $k$ is the half size of the neighbourhood and is fixed for all the points. Note that the saliency measurement in 2.21 is given the name Central Saliency because the point for which the saliency is calculated lies in the middle of the curve segment $AB$. In fact other $2k$ number of saliency measures can be obtained for each point when the chord shifts its position while keeping the point $C$ to lie inside the neighbourhood of $2k + 1$ points. Since there are $2k + 1$ saliency measures for each point, one can either choose a particular one of them (like Central Saliency) or use a combination of them. Figure 2.5 shows the central and maximum saliency as well as three types of combined saliency, which are calculated by using the median, the average and the gaussian weighted sum of the $2k + 1$ measurements.

2.3.2 Criteria for Choosing Saliency Measurements

The saliency measurement proposed above is affected by two types of noise. One is due to the quantisation noise of the pixel positions, as illustrated in the image in Figure 2.4 and in the plot of the saliency in Figure 2.5. The other source of noise is due to the genuine grey level noise in the image that affects the accuracy of the edge detection, which can be assumed to be Gaussian in nature.

In Section 2.1, we have presented a Computational Theory of Feature Detection which states that a signal is optimal for feature detection if and only if its signal amplitude and cur-
2.3. SALIENCY MEASUREMENTS THAT ARE NOT BASED ON ANGLES OR CURVATURES

Figure 2.3: Saliency as the distance to a fixed length chord

Curvature at the salient points are maximised relative to the noise level. The theory gives us direct objective measures to compare different types of saliency measurements. Measurements of the signal amplitude, curvature and noise power can be performed either theoretically or experimentally.

It is possible for us to tell exactly the noise power in the output of our filter relative to the noise power in the raw image given the details of the filtering process and model of the input noise. There are well known methods of analysis of the noise suppression performance of linear and median filters. In our case we may consider a Gaussian Additive noise source for edge detection and a locally uniformly distributed noise source for quantisation. One can also estimate the noise level experimentally by measuring the average output power at the zero curvature region in our sample data. This approach is more advantageous since it does not require the theoretical assumption about the noise, and therefore is more realistic when dealing with complex natural images. The disadvantage is that this method requires the knowledge about the ground truth of the sample images. However for the purpose of comparing the performance of different saliency measurements, we can select our data with known ground truth.

Measurement of signal curvature can be done locally by a fixed mask. As this measurement is used purely for comparison purposes, the exact curvature value is not needed. As long as the curvature is measured the same way across all the different saliency measurements considered, we can compare their relative performance in terms of localisation. Comparing the height of the peaks and the levels in the flat regions also gives a good indi-
2.3. Saliency Measurements That Are Not Based on Angles or Curvatures

Local Peak Saliency

Local Average Saliency

Gaussian Saliency

Local Median Saliency

Central Saliency

Figure 2.4: Comparison of various saliency measurements (the graphs measure the saliency along the contour of the spanner shape in Figure 2.5)
cation of the signal to noise ratio. Obviously a saliency measurement that gives the highest product of the SNR and localisation measures is the optimal one.

From Figure 2.4, we can see that the central saliency provides the best signal to noise ratio and localisation around its peaks. This is because the \( 2k + 1 \) saliency measurements for each point are heavily correlated both on its signal and noise component. Therefore the ensemble average, or median filter does not enhance the signal to noise ratio but only weakens the signal.

2.4 Smoothing Using an Optimal FIR Filter

In the previous two sections we introduced methods for obtaining curvature and saliency measurements from edge chains. This and the next section are concerned with filtering the saliency or curvature measurements in order to detect the points of interest.

Asada and Brady [2] used a gaussian filter to smooth the orientation data in order to compute curvature. Our method draws on their approach but there are two major differences:

1. Instead of using a gaussian filter, we used the criteria proposed by Spacek and the result of Petrou to obtain the optimal shape of the FIR filter. This filter is optimised
2.4. SMOOTHING USING AN OPTIMAL FIR FILTER

for good detection, localisation and suppression of false responses in detecting ramp shaped angle changes. The optimal filter for detecting inflexion points can be derived with the same process when the profile of angles at the inflexion point is modelled by a gaussian function indicating an extremum of angle.

2. In Section 2.2, we have shown how to estimate the noise of the angle measurement. In the followings, we show that the filter size \(w\) in our method can be adapted according to the local estimate of the noise.

2.4.1 The optimal filter for ramp angle changes

A curvature extremum is characterised by an abrupt angle change. Here we model the profile of this angle change with a ramp function:

\[
g(x) = \begin{cases} 
1 - e^{-sx}/2 , & x \geq 0 \\
e^{sx}/2 , & x \leq 0
\end{cases}
\]  

(2.22)

In principle the detection of curvature extrema follows the same rule that guides the design of edge detector (which is essentially the detection of gradient extrema in 2D space). Canny formulated three separate criteria for good extrema detection for 1D signal \(f(x)\), namely the good detection criterion (maximum SNR):

\[
S = \frac{\int_{-w}^{w} g(x)f(-x)dx}{E(\eta^2)\sqrt{\int_{-w}^{w} |g(x)|^2dx}}
\]  

(2.23)

localisation criterion:

\[
L = \frac{s^2\int_{-w}^{w} g'(x)f'(-x)dx}{E(\eta^2)\sqrt{\int_{-w}^{w} |g'(x)|^2dx}}
\]  

(2.24)

and suppression of false response:

\[
C = \frac{1}{w} \sqrt{\int_{-w}^{w} |g''(x)|^2dx}
\]  

(2.25)

Spacek has given a combined criterion function for a step signal. Petrou extended this work for the detection of ramp signal of the form of Equation 2.22 (assuming symmetrical filter
2.4. SMOOTHING USING AN OPTIMAL FIR FILTER

\[ P = \frac{s^4 \int_{-w}^{0} g(x)(1 - e^{tx})dx \int_{-w}^{0} g(x)e^{tx}dx|^2}{w^2 \int_{-w}^{0} |g(x)|^2 dx \int_{-w}^{0} |g''(x)|^2 dx} \quad (2.26) \]

Following the derivation by Petrou [13], we have the optimal filter function:

\[ f(x) = e^{Ax}[L_1 \sin(Ax) + L_2 \cos(Ax)] \]
\[ + e^{-Ax}[L_3 \sin(Ax) + L_4 \cos(Ax)] + L_5 x + L_6 e^{tx} + L_7 \quad (2.27) \]

where \( L_1 \ldots L_7 \) have been optimised numerically for different filter size \( w \) [13]. The optimisation can be done once off-line and the results are recorded into a filter table for later usage. Note that Equation 2.27 was originally derived for optimal edge detection. Since we can use the same criterion function to differentiate the quality of our curvature extrema detectors that are based on FIR filtering, Equation 2.27 in its 1D form can be used directly as an optimal detector for curvature extrema with ramp-like angle change.

### 2.4.2 Determining filter size \( w \)

Many researcher have encountered the problem of choosing an optimal filter size \( w \) to simultaneously maximise both SNR and localisation criteria (Equation 2.23 and 2.24) [3] [13]. A large filter size normally gives a better signal to noise ratio, a small filter size can improve the accuracy of locating the peak. In the problem of edge detection, Canny argued that since the signal-to-noise ratio of each edge is likely to be different in an image, the size of the operator should be adjusted dynamically by the algorithm and this requires a local estimate of the noise energy in the region surrounding the candidate edge[3]. Similarly, in the problem of detecting curvature extrema we have shown that the noise power of the angle measurement is a point-dependent function (Equation 2.19). Thus we should make our filter size adjustable according to local angular noise estimation given by Equation 2.19.

Observing the fact that the strength of edge is likely to be constant along a curve for certain length, and jumps to another value at another edge or after a surface intersection, we may assume that within a certain region the angular noise is stationary. So for each edge segment where edge pixels have a similar gradient, we may begin searching for a filter size \( w \) by assuming \( E(\eta^2) \equiv \overline{E(\eta^2)} \) to be constant. \( \overline{E(\eta^2)} \) is the average of m.s. angular errors estimated by Equation 2.19.
In principle we can solve Equation 2.23 for the value of $w$ so that the detection criterion $S$ (which is the signal-to-noise ratio) equals a given constant. On the other hand, we may want to find an appropriate $w$ from Equation 2.24 to satisfy a specification of localisation. In both cases, the $w$ found might not be a global optimum. If we attempt to optimise Equation 2.26 directly, we will find that the larger $w$ is the greater $P$ is. But clearly, a large smoothing kernel is not always desirable, as it would perform poorly in terms of localisation. This is because the larger the filter the more the edge “looks” to the filter like an ideal step edge [13]. According to our localisation criterion $L_g$ in Equation 2.2, large smoothing kernels tend to reduce the curvature of the signal, thus making non-maxima suppression in the region more difficult. Since both detection and localisation criteria (Equation 2.23 and Equation 2.24) when treated as functions of $w$ are generally monotonic, we can solve either one of the two equations for $w$ using logarithmic search. As the number of permissible values of $w$ is rather limited, the search for an optimal $w$ can be quite fast (typically less than 5 iterations to converge).

Now we summaries our curvature feature detection algorithm:

1. Perform a separable optimal filtering edge detection. Record angle and gradient information and the estimate of grey level noise. Remove trend signal from the angle measurements.

2. Split edge strings into segments so that in each segment the variance of the gradient is less than or equal to a given constant $G$. Within each segment, the filter size will be constant.

3. For each segment:
   - perform a logarithmic search for optimal filter size $w$ which satisfies a specification either in terms of criterion $S$ in Equation 2.23 or criterion $L$ in Equation 2.24;
   - select filter weights from the filter table, and generate the filter mask. Convolve the mask with the angle data;
   - differentiate the filtered result angle data to get curvature estimates;

4. Locate extrema by thresholding and non-maxima suppression.
2.5. SMOOTHING USING ADAPTIVE FILTERS

In this section we present two adaptive filters for the task of detecting curvature features. The motivation for using an adaptive filter for this task is that we are interested in recovering the sharp peaks of the saliency measure. A fixed low pass filter, no matter how it is optimised, will always tend to attenuate both the noise and the sharp signal peaks in the waveform, as long as they both appear in the higher band of the spectrum of the data. To avoid over smoothing of the signal at peaks, the filter needs to be adaptive. According to our computational theory of feature detection introduced in Section 1, peaks in the signal are characterised by points that have high Local Signal to Noise Ratio $|\frac{f(x)}{E(n^2)}|$ and high Local Curvature to Noise Ratio $|\frac{f''(x)}{E(n^2)}|$. The heuristic behind our adaptive smoothing schemes is to provide less smoothing when these two measurements are high, and more smoothing when these measurements are low, thereby reducing the noise content in the data while preserving the high curvature at the points of interest.

Consider that we calculate our filter output iteratively, and the $i$th filter output is $f_n^i$. First we note that the fact that the most important information in our saliency measurement is contained largely in the lower band of the signal spectrum, therefor we also assume that the spectral characteristic of our adaptive filter will vary between a narrow band low-pass filter and an all-pass-filter. This low-pass filtering will always tend to reduce the rapidly changing components (i.e. the higher order differential terms) of the signal, i.e. $f_n^{i+1} \leq f_n^i$, and $f_n^{ii+1} \leq f_n^{ii}$. Therefore given a specified noise suppression capability, an optimal filter will minimally reduce $f_n^i$ and $f_n^{ii}$ at the peak points, thereby maximising our criteria $|\frac{f(x)}{E(n^2)}|$ and $|\frac{f''(x)}{E(n^2)}|$ at the salient points in the final output of the filter. In other words, an optimal filter should maximally preserve the sharp peaks in the signal.

If we apply Taylor expansion of the signal around point $n$, we will see that signal value $f_x$ at point $x$ which is in the neighbourhood of $n$ can be calculated by

$$f_x = f_n + f'_n \delta + f''_n \delta^2 + O(\delta^3) \quad (2.28)$$

Note that the terms $f_n$ and $f''_n$ are exactly what we need to maximise given a specification of noise suppression capability. Also since the noise is characterised by a rapid small fluctuation in the signal, filtering the signal by reducing the term associated with $f'_n$ will signif-
2.5. SMOOTHING USING ADAPTIVE FILTERS

icantly reduce the noise component. This however will not affect the terms associated with \( f_n \) and \( f'_n \). There are many ways to express the optimisation goal in term of a single valued function which our filter will be maximising during each iteration. For example function

\[ P = \sum (f_n^2 + k_2 f'_n^2 - k_1 f_n^2) \]  

(2.29)

where \( k_1 \) and \( k_2 \) are positive numbers, would be a suitable formalisation of the optimisation goal.

In [17], Saint-Marc et. al. proposed an adaptive filter for smoothing the saliency data. First we will introduce this filter and show that it is in fact an iterative filter embedded with an optimisation goal to reduce the components associated with \( f'_n \) in the signal. Secondly we will introduce a new adaptive filter based on a more complete set of optimisation goals according to our feature detection theory. Our new filter is amazingly simple and delivers a performance superior to all previously introduced filters in terms of noise suppression and peak preservation ability.

2.5.1 Filter Proposed by Saint-Marc et.al.

In this subsection, we describe the filter proposed by Saint-Marc et.al. [17] for feature detection problems. Let \( f^i_n \) be the saliency measure at point \( n \) during \( i \)th iteration. \( f^i_n+1 \) is given by:

\[ f^i_n+1 = \frac{1}{\sum_{j=-1}^{+1} W^i_{n+j} f^i_{n+j}} \sum_{j=-1}^{+1} W^i_{n+j} f^i_{n+j} \]  

(2.30)

where the set of filter weights \( W^i_n \) is an array of the same length as the data. The filter weights \( W^i_n \) are calculated from the \( i \)th estimate of the gradient \( f^i_n \),

\[ W^i_n = \exp(-\frac{(f^i_n)^2}{2\sigma^2}) \]  

(2.31)

where \( \sigma \) is the overall width of the filter and is kept as a constant according to the original design by Saint-Marc et.al. The operation of this filter can be explained by the following heuristics. When \( \sigma \) is large, the filter provides maximum smoothing and resembles an average filter. When \( \sigma \) is small, every large discontinuity will cause the filter to stop diffusion and no smoothing will take place. This can be neatly explained by our feature detection
2.5. SMOOTHING USING ADAPTIVE FILTERS

theory. Since 2.31 gives a small filter weight to point \( n \) whenever the derivative at the point \( f'_n \) is large, this filter is reducing the 1st order components in the overall signal, i.e. it has an objective function similar to

\[
P = \sum (-k_1 f'_n^2)
\]

(2.32)

One can see that this is only one of the terms in the objective function expressed in 2.29. In the next section, we will develop a filter that encompasses a more complete set of criteria as given by our feature detection theory.

The outputs of a Saint-Marc filter are shown in Figures 2.6, 2.7, 2.8 and 2.9 for various noise conditions. One can see from these figures that the filter proposed by Saint-Marc et al. generally outperforms non-adaptive filters in terms of Signal to Noise Ratio and Localisation. In particular, its noise suppression capability is much better than non-adaptive filters due the elaborate design aimed at suppressing rapid signal changes (measured by the 1st order derivative of the signal).

2.5.2 Our Adaptive Filter

We have shown that Saint-Marc filter only partially optimises the objective function given in 2.29. Now we are going to design a fully optimised adaptive low-pass filter that maximally preserves \( f_n \) and \( f''_n \), while reducing \( f'_n \).

A simplification can be made to our objective function by noting that an arbitrary function can always be represented as the sum of an even function and an odd function \( f_x = e_x + o_x \) centred at point \( x \) (the centre of even/odd symmetry is situated at \( x \)). When Taylor expansion is applied to \( f_x \) at point \( n \) near \( x \), the even order differential terms define the even function \( e_x \) and odd order differential terms define the odd function \( o_x \). Ignoring the higher order terms in 2.28, we see that the even component \( e_x \) contains objective \( f_n \) and \( f''_n \) in the summation form \( f_n + f''_n \delta^2 \), and the odd component \( f'_n \delta \) is the term that is minimised by the Saint-Marc filter. Therefore we can write our objective function as follows:

\[
P = \sum_{x=n-m}^{n+m} (e_x^2 - o_x^2) = \sum_{y=-m}^{m} \left( \frac{(f_{n+y} + f_{n-y})^2}{2} - \left( \frac{f_{n+y} - f_{n-y}}{2} \right)^2 \right) = \sum_{y=-m}^{m} f_{n+y} f_{n-y}
\]

(2.33)

where \( m \) is the size of the neighbourhood. In practice, \( m = 1 \) is usually chosen. This keeps the measurement as local as possible to ensure a good localisation. The influences between
more distant neighbours are propagated through multiple iterations.

Having constructed the objective function. We now give the filter weights for the case of $m = 1$:

$$f_{n+1}^i = W_{-1}^i f_{n-1}^i + W_0^i f_n^i + W_1^i f_{n+1}^i$$  \hspace{1cm} (2.34)

where filter coefficients $W_j$ are given by

$$W_0^i = \frac{f_{n+1} f_{n-1}}{3\sigma} + \frac{2}{3}$$  \hspace{1cm} (2.35)

$$W_{-1}^i = W_1^i = \frac{1 - W_0^i}{2}$$  \hspace{1cm} (2.36)

This filter adjusts its shape from a single spike in the middle ($W_0 = 1, W_1 = W_{-1} = 0$) which has a uniform spectrum (i.e. no smoothing at all), to a local average filter ($W_1 = W_{-1} = W_0 = 1/3$) which when applied iteratively approximates a narrow-band low pass filter. The result of our filter is compared with other filters in Figure 2.6, 2.7, 2.8 and 2.9. We show that our filter is the most effective for reducing both quantisation and Gaussian noise while delivering the highest response at the peaks in the signal. The noise suppression ability, as can be seen the Figures, is comparable or better than that of Saint-Marc filter. Due to the more sophisticated objective function that contains positive terms of $f_n$ and $f'_n$, our adaptive filter provides the greatest peak response most of the time as compared to other filters, and gives the largest curvature at local maxima and therefore the best localisation and suppression of false maxima.
2.6 Remarks and Conclusion

In this chapter we presented a new theory for feature detection. We argued that criteria for optimal computation of "featured-ness" measurement should be tuned so as to match the characteristics of the thresholding and non-maxima suppression processes. Based on this rationale, we have proposed two criteria for optimal saliency measurement processing 2.1 2.2. The advantages of the new theory are as follows.

1. Our criterion functions provide an objective measurement about how good a signal is for non-maxima suppression and thresholding. The functions are local and easy to measure, and are also independent of the type of processing involved in generating the signal. They can be used directly in the experiments as objective measures to compare the quality of different feature detection algorithms as well as the quality of the raw data. In Section 2.3, we have used these criteria to compare and choose the appropriate saliency measurements.

2. The criteria proposed by us can be directly used to design a wide range of feature detection methods, i.e. they are not just limited to FIR filtering. We have shown how to design adaptive filters that maximise these criteria.

Also we have shown the relationship between our criteria and Canny’s criteria, and the relationship between our criteria and adaptive Saint-Marc filter.

Equation 2.19 gives a point dependent formula of angular noise. When the gradient of edges is relatively fixed in a certain region, we can substitute the error estimates with their average, and provide a method of dynamically determining the appropriate filter size based on the specification of the signal and local noise conditions.

Note also that the optimal FIR filter formula 2.27 is an entirely one dimensional application of the edge detection theory proposed by Canny and Spacek and the analysis by Petrou. Here the problems of edge detection and curvature feature detection from angle data are subject to the same principle and measure of quality.

Finally, we have compared a number of different filters used for feature detection from noisy data, including Median, Optimal FIR, Saint-Marc and our adaptive filter. We demonstrated the potential of the proposed objective function by means of designing a very simple adaptive filter, and showed that this filter has the best all-round performance in the presence
of both Gaussian and quantisation noise. The proposed methodology also offers opportunity and means to design much more sophisticated adaptive filters that explore the utility of the objective functions given section 2.5 (Equation 2.29) and in general the criterion functions given in section 2.1 further.
2.6. REMARKS AND CONCLUSION

Figure 2.6: Comparison of various filters in presence of small quantisation noise
Figure 2.7: Comparison of various filters in presence of large quantisation noise.
Figure 2.8: Comparison of various filters in presence of small Gaussian noise
Figure 2.9: Comparison of various filters in presence of large Gaussian noise
Bibliography


Chapter 3

Point Feature Based Invariant Measurement : Unary and Binary Forms

3.1 Introduction

Transformation invariant object recognition is a problem that has long been pursued by computer vision researchers. Assuming rigid objects, when the scene that contains an object to be recognised is projected onto the imaging plane, the projected image of the object is related to its model by a projective transformation. In this chapter we develop a projective invariant representation for both scene and model in the context of model based vision. The aim of the proposed representation is to facilitate an Attributed Relational Graph (ARG) object matching based on only unary and binary relations.

If we are able to separate the object in the image from the background, recognition can be done by comparing the invariant representations of the object and the model. Many studies have been reported on transformation invariants [7], and how to use them for recognition purpose [21]. However, in many realistic situations, we can rarely isolate the complete object from the background in the presence of clutter and occlusion. Invariant matching based approaches like affine invariant Fourier descriptors and moment invariants [7] [21] may easily fail when applied to cluttered and incomplete data.
One way to get round this problem is to represent both image and the model as collections of local image features and find the partial inexact matching between these two feature sets. Since the features are extracted locally, the collections of them are less vulnerable to the effect of occlusion. Partial matching allows extraneous features to be unmatched, and inexactness allows image features to deviate from the model as a result of noisy data. Mathematically, the problem of image feature matching has been formulated as one of Attributed Relational Graph (ARG) matching [1] [3]. Local image features such as segments of polygonal or spline approximations of the contour [16] [22] [23], colour clusters [6] and regions [17] are often depicted as the nodes of the ARG. The geometric arrangement of these features representing the structure of the image is captured by the relational measurements such as distance, angle, adjacency etc. These relational measurements are associated to the arcs of the ARG image representation. Matching, in this context, refers to a process that finds the correspondences between nodes of the two ARGs, one representing the observed image (the scene), the other representing the model.

An arc of an ARG can connect more than two nodes, in which case it represents a high order relation between the matching entities and the ARG is in fact a hyper-graph. Many researchers have considered transformation invariant constraints being such high order relations, since the transformation is applied globally e.g. [19]. Consequently, it is not only related to any one local shape primitive or any pair of them but to all nodes in the graph (in our case to all features on the same surface patch). Before this work, the constraint that all the matched local features of a rigid shape must undergo the same transformation has always been formulated in terms of a high order relation in the matching problem [18], [20].

Random hyper-graph matching is an NP complete problem. Recently there have been some developments in solving the problem of Attributed Relational Graph matching in polynomial time [1]. In this context relaxation labelling deserves a particular mention [5]. However even now we are still not able to enforce general high order relational constraints in these algorithms because of the high computational cost, with the only exception of objects arranged in a lattice structure where the correspondence problem does not arise.

In this chapter, we show that it is possible to derive an appropriate set of unary and binary measurements capable of representing the global transformation constraint and thus to eliminate the necessity of using high order relations in the first place. The idea is to use
3.2 PROJECTIVE TRANSFORMS

invariant unary and binary relations. The considerable recent effort in developing invariant representations for object recognition is reflected in [12, 22, 8, 9, 10, 11, 13, 14, 15]. The approaches include the use of algebraic and differential invariants. Our objective is to adopt invariants that can be computed locally and to capture the global constraints in terms of binary relations. For this reason we generalise the concept of barycentric coordinates, used in [28] to define affine invariants for geometric hashing, so that it can be applied to any projective transformation. Barycentric coordinates are used as invariant unary relations in our ARG matching approach. We then complement the unary relation representation of objects by invariant binary relations. Specifically, we show that for a pair of nodes in the ARG, the product of the barycentric coordinate system for one node with the inverse of the barycentric coordinate system for the other node is invariant under the projective transformation for which the local coordinate systems have been constructed. The proposed unary and binary invariant relations provide an orthogonal decomposition of shape.

We demonstrate the proposed methodology of projective transformation invariant object representation on several examples. First we illustrate the stability of the shape representation in terms of unary relations both visually and numerically. We then experimentally demonstrated the invariance of binary relations on a star-like object. We show experimentally that the binary relations derived are invariant. The final example demonstrates the proposed approach as a tool for 3D object recognition.

The chapter is organised as follows. In the next section we recollect common projective transformations to which our representation should be invariant. In Section 3.3 we propose unary and binary measurements that are invariant to specified projective transformations. The use of the proposed representation in shape matching is demonstrated in Section 3.4. Section 3.5 offers a summary of the presented work and conclusions.

3.2 Projective Transforms

We commence by demonstrating how to generate a projective invariant ARG representation of object shape. In particular, this representation will only consist of unary and binary measurements.

Formally we define our second order ARG as a quadruple \( G = (N, E, f, g) \), with \( N \) being
a set of nodes and $E \subset \{(i,j)|i \in N, j \in N\}$ being a set of arcs. $f : N \rightarrow X$ is a function over $N$ representing the unary measurements associated with the nodes, where $X$ denotes the vector space of unary measurements; $g : E \rightarrow A$ is a function representing a binary measurement associated with each arc of the graph, and $A$ is the vector space of the binary relational measurements. With $N$ defined purely for labelling purposes, shape information contained in the ARG description manifests itself mainly through the unary and binary measurements. Therefore, our major task is to develop unary and binary measurements that suit the projective invariant requirement.

Here we use edges as our shape representation. Our ARG representation requires the edge strings to be decomposed into coherent segments that represent the nodes for matching. We segment the edge strings at vertices and inflexion points because the position of these points is relatively stable under projective distortion and can be detected locally. In order to generate invariants, we partition the edge strings in such a way that each node will contain a sufficient number of feature points for generating the projective invariant. The resulting edge segments are used as the nodes of the ARG representation.

We represent an arbitrary 2D image point $p = (p_x, p_y)$ with its homogeneous coordinate

$$P = \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix}$$  \hspace{1cm} (3.1)

where $p_x = P_x/P_z$ and $p_y = P_y/P_z$. Therefore we can represent a projective transform from point $p$ to $p'$ with a linear transform between their homogeneous coordinates:

$$P' = PT$$  \hspace{1cm} (3.2)

Since a linear transform in the 3D homogeneous coordinate system corresponds to a projection between two 2D coordinate systems ([7], Chapter 1), matrix $T$ can be used to represent a wide class of 2D projections including affine and perspective transforms [7] [24] [26]. Also it should be noted that our analysis is not limited to the case of 2D projections. 3D point projections can be captured by a 4x4 matrix in a 4D homogeneous coordinate system. To illustrate our approach, we will concentrate on the following four types of 2D projections:

1. **Translation**, which is represented by two parameters $t_x$ and $t_y$ in the transformation
3.2. PROJECTIVE TRANSFORMS

matrix:

\[
T_t = \begin{pmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
t_x & t_y & 1
\end{pmatrix}
\]  

(3.3)

2. Similarity transform, which includes translation, rotation and scaling. It preserves angles and ratio of lengths and has 4 degrees of freedom.

\[
T_r = \begin{pmatrix}
\tau_1 & \tau_2 & 0 \\
-\tau_2 & \tau_1 & 0 \\
t_x & t_y & 1
\end{pmatrix}
\]  

(3.4)

A subset of the similarity transform is called the rigid (Euclidean) transform, where the scaling factor is 1, i.e.

\[
r_1^2 + r_2^2 = 1
\]  

(3.5)

3. Affine transform, which includes scaling and shear. It is represented by

\[
T_a = \begin{pmatrix}
\tau_1 & \tau_2 & 0 \\
\tau_3 & \tau_4 & 0 \\
t_x & t_y & 1
\end{pmatrix}
\]  

(3.6)

where \( \tau_1, \tau_2, \tau_3 \) and \( \tau_4 \) together describe rotation, scaling and shear. It preserves parallelism as well and has six degrees of freedom.

4. Perspective transform, which includes the single projection point transform, 2 point perspective transform in 2D and 3D, and 3 point perspective transform in 3D. In the case of 2D 2 point perspective transform (which has 8 degrees of freedom), transformation matrix \( T \) is usually written as :

\[
T_p = \begin{pmatrix}
\tau_1 & \tau_2 & p \\
\tau_3 & \tau_4 & q \\
t_x & t_y & 1
\end{pmatrix}
\]  

(3.7)

where \( p \) and \( q \) are inversely proportional to the distance of the projection centres to the
imaging plane.

By analogy, the forthcoming analysis can also be extended to a more general case of 3D to 3D photographic projections. However, as we are only dealing with 2D images, this will not be shown. In the following section we will derive the unary and binary measures for each of the above cases and compare our approach with the conventional methods.

3.3 Projective Invariant Unary and Binary Measures

3.3.1 A general form of projective invariant unary and binary measurements based on point features

In this section we develop our binary and unary measurements based on a set of reference points. These points are supposed to be detectable under projective transforms described above. For example, corners (large curvature extrema) as well as bi-tangent points around concavities [27] are stable under all these transforms. Given a set of reference points, it is possible to generate a basis matrix:

\[
B = \begin{bmatrix}
B_{11} & B_{12} & \ldots & B_{1n} \\
B_{21} & B_{22} & \ldots & B_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
B_{m1} & B_{m2} & \ldots & B_{mn}
\end{bmatrix}
\] (3.8)

Provided the basis matrix \(B\) is square and non-singular, we can define the barycentric coordinate of image point \(P\) with basis \(B\) as

\[
C_B(P) = PB^{-1}
\] (3.9)

Suppose we apply an arbitrary projective transform \(T\) to both, point \(P\) and the basis, then the barycentric coordinate of the transformed point is:

\[
C_B'(P') = C_B(PT) = (PT)(BT)^{-1} = PTT^{-1}B^{-1} = PB^{-1} = C_B(P)
\] (3.10)
3.3. PROJECTIVE INARIANT UNARY AND BINARY MEASURES

Therefore we conclude that the barycentric coordinate is invariant to arbitrary projective transform $T$ if

1. one can find a basis matrix $B$ for an arbitrary image without the knowledge of the projection, i.e. image features used to construct the basis matrix $B$ must be invariant and detectable under arbitrary projection;

2. $B$ is non-singular;

3. basis $B$ from one image and the corresponding basis $B'$ obtained for the projection of the image are related to each other by the multiplication of the transformation matrix $T$, i.e. $B' = BT$. In group representation theory terms, $T$ is a representation of the underlying transformation group and $B$ is transformed as a vector.

The first condition is presumably satisfied when we generate a basis matrix using salient image features that are invariant to the projection, e.g. colours, sharp corners, bi-tangent points etc. In the next subsection, we will give a basis matrix for each type of projection that satisfies condition 2 and 3. A detailed proof for the last two conditions for each proposed basis can be found in [30].

We shall now construct unary and binary relations using the basis representation. Suppose a node (a local shape primitive) $V_i$ contains $n$ points $(P_1, \ldots, P_n)$, and the basis of the node is $B_i$. The barycentric coordinates of these points (together with the measurements such as colour, intensity associated with these points) can be used jointly as the unary measurements of the node:

$$X_i = (C_{B_i}(P_1), C_{B_i}(P_2), \ldots, C_{B_i}(P_n))$$

(3.11)

If for each $j$, $C_{B_i}(P_j)$ are calculated from a common basis $B_i$, then $X_i$ describes the shape of the segment irrespective of the projection it undergoes. Note that for 2D transforms, the basis matrices are 3x3, and our invariant measurements $C_{B_i}(P_j)$, $j = 1 \ldots n$ are 3 dimensional vectors. However the 3 components of each $C_{B_i}(P_j)$ are not independent from each other. In practise, only 2 out of these 3 numbers are needed to describe the point. We will show this point later after we derive the bases for various transformations.
3.3. PROJECTIVE INVARIANT UNARY AND BINARY MEASURES

For binary measurements, we use the barycentric coordinate of the basis $B_i$ of node $V_i$ in relation to the basis $B_j$ of the node $V_j$ i.e.

$$A_{ij} = B_i B_j^{-1} \quad (3.12)$$

It can be proved that this measurement is also an invariant to arbitrary transformation $T$ if both bases are measured from the same rigid object whose features undergo the same transformation. Both unary and binary features are derived from the same basis matrix. $X_i$ describes the intra-node features and $A_{ij}$ represents the inter-node relations. $X_i$ and $A_{ij}$ together provide an orthogonal decomposition of the shape data into binary and unary measures.

The global constraint that requires all the matched nodes to undergo the same transformation is embedded in the binary measure $A_{ij}$ when we expand the neighbourhood of the node $V_i$ to all $V_j \forall j \neq i$ in the model. Suppose two nodes $V_i$ with $B_i$ and $V_j$ with $B_j$ in model $G$, correspond to two nodes $V'_p$ with $B'_q$ and $V'_q$ with $B'_q$ in scene $G'$. It is easy to see that the matched pairs $(V_i, V'_p)$ and $(V_j, V'_q)$ respectively determine two possible underlying transformations $T_{ip}$ and $T_{jq}$ from the model to the scene:

$$B'_p = B_i T_{ip} \quad (3.13)$$
$$B'_q = B_j T_{jq} \quad (3.14)$$

From 3.13, 3.14 and 3.12, we see that if one observes binary measurement $A_{ij}$ between $V_i$ and $V_j$ and $A'_{pq}$ between $V'_p$ and $V'_q$ then

$$A'_{pq} = B'_p B'_q^{-1} = B_i T_{ip} (B_j T_{jq})^{-1} = B_i T_{ip} T_{jq}^{-1} B_j^{-1} = A_{ij}, \text{ iff } T_{ip} = T_{jq} \quad (3.15)$$

$$T_{ip} = B_i^{-1} B'_p = (A_{ij} B_j)^{-1} A'_{pq} B'_q = B_j^{-1} A_{ij}^{-1} A'_{pq} B'_q = T_{jq}, \text{ iff } A'_{pq} = A'_{ij} \quad (3.16)$$

Equations 3.15 and 3.16 tell us the fact that if the binary measurements in the two images are the same (i.e. $A'_{pq} = A_{ij}$), so should the underlying transformation between the pairs of features i.e. $T_{ip} = T_{jq}$, and vice versa. Also if $T_{ip}$ and $T_{jq}$ are different, our binary measures from the two image must be different too. Therefore binary measurement $A_{ij}$ expresses the
compatibility of a pair of labelling with respect to the common projective transform constraint.

Equations 3.11 and 3.12 have provide a general framework for deriving invariant measurements. Now we will demonstrate how to obtain the appropriate basis matrix from a given set of reference points.

### 3.3.2 Basis matrices for various projections

For translation invariant matching (Equation 3.3), conventional approaches typically involve correlation measures. For the ARG matching based approach, measurements like direction and distance are usually used. In our representation, we only need one reference point as a basis since the total number of degrees of freedom is two. Let the reference point be \((a, b)\).

We fill the unoccupied two row vectors of the basis matrix with two orthogonal vectors:

\[
B_t = \begin{pmatrix}
  a & b & 1 \\
  0 & k_1 & 0 \\
  k_2 & 0 & 0 \\
\end{pmatrix}
\]  

(3.17)

When \(k_1 \neq 0\) and \(k_2 \neq 0\) are real constants, it is easy to see that \(B_t\) is always invertible (c.f. Condition 2 above). Now all we need to do in order to prove \(B_t\) is a valid basis is to show that the coordinates of the basis points are translated accordingly while the two orthogonal vectors are unchanged (i.e. Condition 3):

\[
B_tT_t = \begin{pmatrix}
  a & b & 1 \\
  0 & k_1 & 0 \\
  k_2 & 0 & 0 \\
\end{pmatrix} \begin{pmatrix}
  1 & 0 & 0 \\
  0 & 1 & 0 \\
  t_x & t_y & 1 \\
\end{pmatrix} = \begin{pmatrix}
  a + t_x & b + t_y & 1 \\
  0 & k_1 & 0 \\
  k_2 & 0 & 0 \\
\end{pmatrix} = B'_t
\]

(3.18)

which the right hand bracketed term of Equation (3.18) confirms.

Suppose there is an arbitrary point represented by its homogeneous coordinate \((x, y, 1)\).

The invariant representation of the point is

\[
(C_x, C_y, C_z) = (x, y, 1)B_t^{-1} = (1, \frac{y - b}{k_1}, \frac{x - a}{k_2})
\]

(3.19)

Since \(C_x\) is constant, we choose \((C_y, C_z)\) as our invariant measurements.
For the similarity transformation shown in Equation 3.4, we have to determine 4 parameters. Therefore we choose 2 reference points \((a, b)\) and \((c, d)\) and make the basis matrix:

\[
B_s = \begin{pmatrix}
    a & b & 1 \\
    c & d & 1 \\
    \alpha(a + d - b - c) & \alpha(a + b - d - c) & 0
\end{pmatrix}
\]  

(3.20)

where \(\alpha \neq 0\) is a real number. Expanding \(B_s T_s\), again we can prove this to be a correct basis. For arbitrary 2D point \((x, y)\) we can calculate its barycentric coordinate with respect to basis \(B_s : (C_x, C_y, C_z) = (x, y, 1)B_s^{-1}\). It is easy to show that \(C_x + C_y = 1\), therefore we chose \((C_y, C_z)\) as our similarity invariant representation of point \((x, y)\). One can also prove that basis matrix \(B_s\) is always invertible provided point \((a, b)\) and \((c, d)\) do not overlap. The proof of Condition 3 is as follows:

\[
B_s T_s = \begin{pmatrix}
    a & b & 1 \\
    c & d & 1 \\
    \alpha(a + d - b - c) & \alpha(a + b - d - c) & 0
\end{pmatrix} \begin{pmatrix}
    r_1 & r_2 & 0 \\
    -r_2 & r_1 & 0 \\
    t_x & t_y & 1
\end{pmatrix} = B_s'
\]  

(3.21)

\[
\begin{pmatrix}
    ar_1 - br_2 + t_x & ar_2 + br_1 + t_y & 1 \\
    cr_1 - dr_2 + t_x & cr_2 + dr_1 + t_y & 1 \\
    \alpha(r_1(a + d - b - c) + r_2(c + d - a - b)) & \alpha(r_1(a + b - c - d) + r_2(a + d - b - c)) & 0
\end{pmatrix}
\]  

(3.22)

To date, most relational recognition systems based on matching relational structures are only able to handle rigid transformations. The reason for this essentially lies in the problem of handling global transformation constraints, and in the lack of systematic approaches that use the knowledge of invariants to develop relational representation. In [1] a set of rigid transform invariant unary and binary measures has been proposed for matching straight line segments. These features were chosen largely by heuristic guesses. There is no guarantee of the data being orthogonally decomposed into relational measurements of different orders, and there is no clear indication of how these features can be used for non line segment matching scenarios. In contrast, we have demonstrated that from the generic concept
of "barycentric coordinate", one can derive a large family of unary and binary invariant measures with provable properties. These invariants come with tidy matrix formulations that are easy to program and verify. More importantly, the development of our invariant representations has shown a way of systematic exploitation of invariant knowledge in the context of relational image representation and matching. This will become more clear after we develop our affine and perspective invariant representations.

For the affine transform shown in (3.6), given 3 none collinear reference points \((a, b), (c, d)\) and \((e, f)\), the proposed basis matrix is:

\[
B_a = \begin{pmatrix}
    a & b & 1 \\
    c & d & 1 \\
    e & f & 1 \\
\end{pmatrix}
\]  

(3.24)

The validity of Condition 2 is confirmed by noticing that \(B_a\) is always invertible when the 3 reference points in the basis are not collinear. Also we can prove that \(B_a\) is a valid basis for affine transformation by expanding \(B_aT_a\) and show that the basis matrix undergoes the same transformation as arbitrary point \((x, y)\) does. In fact, since each reference point is transformed independently by \(T_a\) and \(B_a\) is arranged as an ensemble of coordinates of the three reference points, condition \(B'_a = B_aT_a\) always holds.

Substituting (3.24) and (3.1) into (3.9) and simplifying, we obtain an affine invariant representation for arbitrary point \((x, y, 1)\) in the image: (relative to an ordered set of reference points)

\[
(C_x, C_y, C_z) = \frac{\begin{pmatrix} x & y & 1 \end{pmatrix}}{ad + cf + eb - af - cb - ed} \begin{pmatrix} d - f & f - b & b - d \\ e - c & a - e & c - a \\ cf - ed & eb - af & ad - cb \end{pmatrix}
\]

(3.25)

The basis of the form in (3.24) has been proposed in [28] as a feature for geometric hashing and some also used it for matching by back projection. Here we use it as a tool for developing an affine invariant relational description. Note that \(C_x + C_y + C_z = 1\), therefore we choose \((C_y, C_z)\) as our invariant representation of a point.

Perspective transform in (3.7) is another important projection model. For the 2D 2 point perspective transform, we need four reference points \((a, b), (c, d), (e, f)\) and \((g, h)\) to deter-
mine a set of 8 parameters. In order to facilitate calculations, we propose to use 4D homogeneous coordinates of a point instead of the 3D homogeneous coordinates, i.e. to represent a 2D point \((a, b)\) using coordinate \((a, b, 1, 1)\). Accordingly, we augment the perspective transform matrix in (3.7) with a column vector and a row vector:

\[
T' \begin{pmatrix}
    r_1 & r_2 & p & 0 \\
    r_3 & r_4 & q & 0 \\
    t_x & t_y & 1 & 0 \\
    0 & 0 & 0 & 1
\end{pmatrix}
\]

(3.26)

One can show that the 4D homogeneous representation of points and the augmentation of the perspective transform matrix do not affect the calculation of the perspective transform. Consider a point \((x, y)\) perspectively transformed by \(T_p\). Its transformed coordinate will be:

\[
\begin{pmatrix}
    x \\
    y \\
    1
\end{pmatrix}
\begin{pmatrix}
    r_1 & r_2 & p \\
    r_3 & r_4 & q \\
    t_x & t_y & 1
\end{pmatrix}
= \begin{pmatrix}
    r_1 x + r_3 y + t_x \\
    r_2 x + r_4 y + t_y \\
    px + qy + 1
\end{pmatrix}
\]

(3.27)

This calculation in its augmented 4D homogeneous form is:

\[
\begin{pmatrix}
    x \\
    y \\
    1
\end{pmatrix}
\begin{pmatrix}
    r_1 & r_2 & p & 0 \\
    r_3 & r_4 & q & 0 \\
    t_x & t_y & 1 & 0 \\
    0 & 0 & 0 & 1
\end{pmatrix}
= \begin{pmatrix}
    r_1 x + r_3 y + t_x \\
    r_2 x + r_4 y + t_y \\
    px + qy + 1 \\
    1
\end{pmatrix}
\]

(3.28)

From (3.28), we can see that the 4th dimension in the \((x, y, 1, 1)\) representation of point \((x, y)\) does not affect the calculation of the transform. The purpose of augmenting the coordinate system with an extra dimension is to aid the representation of the four reference points in
3.3. PROJECTIVE INVAR IANT UNARY AND BINARY MEASURES

terms of a square and invertible basis matrix, which can then simply be :

\[
B_p = \begin{pmatrix}
  a & b & 1 & 0 \\
  c & d & 1 & 0 \\
  e & f & 1 & 0 \\
  g & h & 1 & 1 \\
\end{pmatrix}
\]

(3.29)

To prove Condition 3 for the basis to be valid, expand \(B_p T_p\) and let the result be \(B'_p\)

\[
B'_p = B_p T_p = \begin{pmatrix}
  a' & b' & z_1 & 0 \\
  c' & d' & z_2 & 0 \\
  e' & f' & z_3 & 0 \\
  g' & h' & z_4 & 1 \\
\end{pmatrix}
\]
\[
= \begin{pmatrix}
  a & b & 1 & 0 \\
  c & d & 1 & 0 \\
  e & f & 1 & 0 \\
  g & h & 1 & 1 \\
\end{pmatrix}
\begin{pmatrix}
  r_1 & r_2 & p & 0 \\
 r_3 & r_4 & q & 0 \\
 t_x & t_y & 1 & 0 \\
 0 & 0 & 0 & 1 \\
\end{pmatrix}
\]

(3.30)

We can easily show that \((a', b', z_1), (c', d', z_2), (e', f', z_3)\) and \((g', h', z_4)\) are the homogeneous coordinates of the projections of the reference points \((a, b), (c, d), (e, f)\) and \((g, h)\) respectively. Thus when the reference points in the model correspond to the reference points in the scene, the basis matrices in the two domain are related to each other by a simple multiplication via the transformation matrix \(T_p\). Also note that the proposed basis \(B_p\) is always invertible when every triplet out of the four points is not collinear. Therefore \(B_p\) satisfies all the conditions of a valid basis.

In summary, we have shown how to use barycentric coordinates and reference points to develop unary and binary projective invariants. The advantages of our approach are as follows :

1. We derived a set of invariant measures from the generic concept of barycentric coordinate. And provided a set of conditions that must be satisfied in order to use the invariant formula. We have demonstrated its applicability in a wide class projective transforms, from translation to perspective. The methodology can be extended to 3-D and photographic projection as well.

2. Invariant measures we developed have a number of desirable qualities. They decompose the data orthogonally into binary and unary relations. It is simple and elegant to allow efficient computation and easy analysis.
3. Most importantly, we have demonstrated a way of capturing global geometric transformation constraint using only binary relations and thus enable the use of deterministic ARG matching algorithms for shape matching under complex projective transforms.

4. Our approach is based solely on salient reference points. Using the proposed unary invariant measurements, we can represent and reconstruct arbitrary measurements associated a node, even it is a unparameterised contour or a group of scattered regions. The nodes of our ARG representation can virtually be anything, not just line segments.

3.3.3 Measurement Error and Stability

Given a model of the error process affecting the measurement of an image data point \( P = (P_x, P_y) \), we can infer the appropriate error model of its invariant representation. We shall illustrate how to derive the noise characteristics of our invariant binary and unary measures for the affine invariance case. The same analysis can be applied to obtain the stability of other projective invariant measures.

We shall assume that the position measurements of all image points are i.i.d. Gaussian with covariance matrix:
\[
\Sigma = \begin{pmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{yx} & \sigma_y^2 \end{pmatrix}
\]

For affine invariants with basis (3.24), we can derive the covariance structure of \((C_x, C_z)\) in (3.25):
\[
\sigma_{cx}^2 = \frac{1}{\Delta^2} (x_1^2 \sigma_y^2 + y_1^2 \sigma_x^2 - 2x_1y_1 \sigma_{xy})
\]
\[
\sigma_{cy}^2 = \frac{1}{\Delta^2} (x_2^2 \sigma_y^2 + y_2^2 \sigma_x^2 - 2x_2y_2 \sigma_{xy})
\]
\[
\sigma_{cx}^2 = \frac{1}{\Delta^2} (-x_1x_2\sigma_y^2 - y_1y_2\sigma_x^2 + (x_1y_2 + x_2y_1) \sigma_{xy})
\]

where \( \Delta = y_1x_2 - y_2x_1 \) is twice the area of the basis triangle formed by the three constituent points, and \( x_1 = a - c \), \( x_2 = e - c \), \( y_1 = b - d \) and \( y_2 = f - d \). It can be easily seen from (3.32),(3.33) and (3.34) that the larger the area of the basis triangle the more stable are the invariant measures. By defining and minimising the average sensitivity and maximal sensitivity of arbitrary point \( P \) to basis \( B \) [28], Gotsman showed that the triangles inscribed in a convex region with a maximum area and triangles with a maximum area/perimeter...
ratio form relatively robust basis. From our point of view, Gotsman's result is a special case that can be derived from (3.32), (3.33) and (3.34) under the assumption that $\sigma_x^2 = \sigma_y^2 = \sigma^2$ and $\sigma_{xy} = 0$. When matching our ARG descriptions of images, the inexactness is captured by modelling the measurement with a probability density function which has a covariance structure described by (3.32)-(3.34).

To verify the measurement stability characteristics of our invariant measurements, we performed controlled experiments as follows:

1. A series of 100 by 100 image are generated, each contains 5,000 random points.
2. Generate a series of triples of points which will form our affine bases. The triplets are selected of different size to illustrate the fact that the noise sensitivity of our invariance measurements depends on greatly on the size of the basis triangle.
3. Perturb positions of both bases and data points with various levels of Gaussian noise and calculate the invariant unary/binary descriptions for each image.
4. The invariant descriptions are mapped back into the image plane using the ground truth of the projection. The average distance of the matched data points are calculated as a indicator of the error generated during the invariance calculation as effect of noise.
5. The matching error as function of noise level and basis size are illustrated in Figure 3.1.

One can see from Figure 3.1 that the stability of the invariant measurements depends greatly on the noise level relative to the basis size. Here we choose to measure the basis size using its area-perimeter ratio, and the unit is in pixels. We also choose the measure the noise level using the standard deviation of point position from its actual position, this also measures in pixels. When the basis size is comparable to the noise level, as illustrated in sub-figure (3.1.1), (3.1.2) and (3.1.3), the measurement error is very large and is highly unstable. This is due to the fact that a badly perturbed small basis can generate great mismatches in the point positions. Therefore it is not possible to perform back-projection base matching in such case, and the compatibility measure for nodes with small basis may appear to be a small and random number. However, when the basis size is larger than 5 times the noise level, the measurement is entirely stable and accuracy improves quickly as the noise level drops.
3.3. PROJECTIVE IN Variant UNARY AND BINARY MEASURES

down and basis size increases. Through extensive controlled experimentation, we obtained
the follow rule of thumb when using the proposed basis under noisy conditions:

- When the estimated error in point positions is greater or comparable to the size of the
  basis, one should not attempt to use back-projection for matching, and the compatibility
  in such situations can be substituted by a small random number. Note that this is
  not a situation that arises often, since in most cases, we can obtain basis triplets that is
  in some order of magnitude greater than the noise level. For example, average corner
detection algorithms generates positional errors (versus the ground truth) under normal
image noise at a level of a few pixels, where as the separations of the corners are
often much larger in the image (often measures in a few tens or hundreds of pixels, i.e.
in most cases the data is in the region described by Figure 3.1.9 or better).

- It is obvious to see from Figure 3.1 that the large the basis the greater stability of the
  measurements with respect to noise. One should try to use larger bases whenever
  possible.

The covariance structure for the proposed invariants of translation, similarity transform
and perspective transform can also be derived in a similar way.

In practice the reference points from which we build the basis are obtained from the image
using some salient features such as inflexions on curves [23], largest inscribed triangles
[28] and bi-tangent points around concavities. As the detection of these points is not noise
free, the basis matrix also suffers from some kind of error, and the effect becomes difficult to
analyse when the error of our invariant representation of the data point $P$ does not only
depend on its positioning error in the Cartesian system but also on that of the reference points.
For the affine basis (3.24), there has been some work directed towards finding the sensitivity
of invariant measurements in relation with the choice of the reference points e.g. in [28].
But in our problem, we are constrained to the choice of reference points which are not only
invariant to the transformation, but also locally detectable and applicable to objects with arbi-
trary convexities and concavities. In practice, we choose a combination of bi-tangent and
maximum average Fischler-Bolles saliency points [25] as the reference points for the above
reason.
3.4 Experiments

In the previous sections we have argued that the image measurements, as measured in the barycentric coordinate system defined by the reference points, are invariant to the projection. We can verify this by visualising the image model and its projection in their barycentric coordinates. Figures 3.2 3.3 3.6 3.7 3.10 3.11\(^1\) show a set of 2D shapes under affine projection, Figures 3.4 3.5 3.8 3.9 3.12 3.13 show the corresponding normalised images in the barycentric coordinate system. One can see that the arbitrary projections do not change the objects’ representation in the barycentric coordinate system. This suggests that the coordinate transform by formula \(P B^{-1}\) (c.f. 3.9) generates invariant unary measures.

Quantitatively, one can demonstrate the invariant properties of the unary relations by plotting the successive values of the samples of a contour in the barycentric coordinate system corresponding to a particular transformation, which is located at a given reference point on the contour. In this simple demonstration the object is a tea pot. We have generated the various views synthetically by re-mapping the original tea pot image shown in Figure 3.4 via the respective transformations. Figure 3.4 shows the example contour from which the invariant data are computed and tested. The invariants computed from this contour for translation, similarity, affine and perspective transforms are shown in Figure 3.16, 3.17, 3.18 and 3.19 respectively. They show a good numerical agreement for all four classes of transformation. It is easy to see that for each case only 2 variables are independent of each other. In the case of translation invariants, one of the 3 parameters is a constant (of value 1, if the basis are chosen with parameters \(k_1 = k_2 = 1\)). For each affine invariant point representation, the sum of the three parameters is 1, so only 2 of them are needed to specify the measurements. Note that the following conditions may contribute to the discrepancies between the model measurements and the scene measurement:

1. Large error in the detection of reference points.

2. Occlusion.

3. Shape shrinkage due to scaling may also reduce the resolution of matches.

\(^1\)Here we use an image coordinate system which is left-handed, and with the origin at the top left corner of the image.
Table 3.1: Comparison of binary measurements $A_{ij}$ of the model and of the scene (for $i = 2 \ldots 5$)

For binary relations we demonstrate the proposed methodology on the star image in Figure 3.20. It contains a group of $n = 10$ feature points from which we construct $n - 1$ nodes with basis $B_i(p_i-1, p_i, p_i+2), i = 1 \ldots n - 1$. The image is matched to its affine transformed version shown in Figure 3.21. For binary measurement, we need to take two sets of reference points from each image, each set contains 3 points for an affine basis. The two sets may share at most two points with each other, therefore in practice, it is sufficient to have an ordered list of four points to obtain a binary measure. Table 3.1 shows that binary relations $A_{ij} = B_iB_1^{-1}, i = 2 \ldots n - 1$ are identical in both images (i.e. they are invariant to affine transform.) Moreover, the binary relations are identical only if the two nodes correspond to each other. The binary relation $A_{1,i}$ never yields the same value for different $i$ unless the nodes completely overlap in the image. This is the case even when the object contains symmetric features (such as nodes $(1,2,3)$ and $(1,10,9)$ etc. in this “star” figure).
3.4. EXPERIMENTS

The last experiment demonstrates a matching problem solved using ARG representations with the proposed perspective invariant binary and unary measures. The scene contains a hexagonal face under a perspective projection in a cluttered background. The system correctly identifies the object of interest using the proposed representation, combined with the relaxation labelling algorithm described in [29].

One of the advantages of the ARG representation of the scene is that the effect of occlusion can be limited to a local region. Provided some nodes in the graph can be matched without occlusion, we can anticipate that by fusing the information from all the nodes in the graph, local discrepancies will be overcome.

<table>
<thead>
<tr>
<th>node triplet $(i-1,i,i+1)$</th>
<th>binary measures from model</th>
<th>binary measures from scene</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(5,6,7)$</td>
<td>0.10 1.14 2.95</td>
<td>0.21 1.17 3.14</td>
</tr>
<tr>
<td></td>
<td>-2.16 -1.27 -2.20</td>
<td>-2.03 -1.19 -2.11</td>
</tr>
<tr>
<td></td>
<td>3.06  1.13  0.25</td>
<td>2.82  1.02  -0.03</td>
</tr>
<tr>
<td>$(7,6,5)$</td>
<td>2.95  1.14  0.10</td>
<td>3.14  1.17  0.21</td>
</tr>
<tr>
<td></td>
<td>-2.20 -1.27 -2.16</td>
<td>-2.11 -1.19 -2.03</td>
</tr>
<tr>
<td></td>
<td>0.25  1.13  3.06</td>
<td>-0.03  1.02  2.82</td>
</tr>
<tr>
<td>$(6,7,8)$</td>
<td>1.14  2.95  1.65</td>
<td>1.17  3.14  1.74</td>
</tr>
<tr>
<td></td>
<td>-1.27 -2.20 -0.75</td>
<td>-1.19 -2.11 -0.69</td>
</tr>
<tr>
<td></td>
<td>1.13  0.25  0.10</td>
<td>1.02  -0.03  -0.05</td>
</tr>
<tr>
<td>$(8,7,6)$</td>
<td>1.65  2.95  1.14</td>
<td>1.74  3.14  1.17</td>
</tr>
<tr>
<td></td>
<td>-0.75 -2.20 -1.27</td>
<td>-0.69 -2.11 -1.19</td>
</tr>
<tr>
<td></td>
<td>0.10  0.25  1.13</td>
<td>-0.05  -0.03  1.02</td>
</tr>
<tr>
<td>$(7,8,9)$</td>
<td>2.95  1.65  2.40</td>
<td>3.14  1.74  2.49</td>
</tr>
<tr>
<td></td>
<td>-2.20 -0.75 -0.05</td>
<td>-2.11 -0.69 -0.03</td>
</tr>
<tr>
<td></td>
<td>0.25  0.10  -1.35</td>
<td>-0.03  -0.05  -1.46</td>
</tr>
<tr>
<td>$(9,8,7)$</td>
<td>2.40  1.65  2.95</td>
<td>2.49  1.74  3.14</td>
</tr>
<tr>
<td></td>
<td>-0.05 -0.75 -2.20</td>
<td>-0.03  -0.69  -2.11</td>
</tr>
<tr>
<td></td>
<td>-1.35  0.10  0.25</td>
<td>-1.46  -0.05  -0.03</td>
</tr>
<tr>
<td>$(8,9,10)$</td>
<td>1.65  2.40  1.00</td>
<td>1.74  2.49  1.00</td>
</tr>
<tr>
<td></td>
<td>-0.75 -0.05  0.00</td>
<td>-0.69  -0.03  0.00</td>
</tr>
<tr>
<td></td>
<td>0.10  -1.35  0.00</td>
<td>-0.05  -1.46  0.00</td>
</tr>
<tr>
<td>$(10,9,8)$</td>
<td>1.00  2.40  1.65</td>
<td>1.00  2.49  1.74</td>
</tr>
<tr>
<td></td>
<td>0.00  -0.05  -0.75</td>
<td>-0.00  -0.03  -0.69</td>
</tr>
<tr>
<td></td>
<td>0.00  -1.35  0.10</td>
<td>0.00  -1.46  0.05</td>
</tr>
<tr>
<td>$(9,10,1)$</td>
<td>2.40  1.00  0.00</td>
<td>2.49  1.00  0.00</td>
</tr>
<tr>
<td></td>
<td>-0.05  0.00  1.00</td>
<td>-0.03  -0.00  1.00</td>
</tr>
<tr>
<td></td>
<td>-1.35  0.00  0.00</td>
<td>-1.46  0.00  0.00</td>
</tr>
<tr>
<td>$(1,10,9)$</td>
<td>0.00  1.00  2.40</td>
<td>-0.00  1.00  2.49</td>
</tr>
<tr>
<td></td>
<td>1.00  0.00  -0.05</td>
<td>1.00  -0.00  -0.03</td>
</tr>
<tr>
<td></td>
<td>0.00  0.00  -1.35</td>
<td>0.00  0.00  -1.46</td>
</tr>
</tbody>
</table>

Table 3.2: Comparison of binary measurements $A_{i,i}$ of the model and of the scene (for $i = 6 \ldots 10$)
3.5 Conclusions

The problem of transformation invariant object recognition was studied. We developed a projective transformation invariant representation for both scene and model which facilitates an Attributed Relational Graph object matching based only on unary and binary relations. The unary and binary measurements used for matching are derived from sets of reference points such as corners and bi-tangent points which are stable under the various transformations considered. Each set of reference points is used to generate a distinct barycentric coordinate basis system associated with one node of the object graph representation. We showed that barycentric coordinates of the reference image points can be made invariant under any arbitrary projective transformation. The conditions that must hold for a basis to be valid were stated. We illustrated how the barycentric coordinate systems can be constructed for the affine and perspective transformations.

For the object and scene representation we use the barycentric coordinates of the reference points, together with auxiliary measurements such as colour and texture as the node's unary measurements. For binary measurements we use the product of the barycentric coordinate system for one node with the inverse of the barycentric coordinate system associated with another node. The unary and binary relations provide an orthogonal decomposition of the shape being matched. They are used in a relaxation process to detect instances of objects consistent with a given model.

We demonstrated the proposed methodology of projective transformation invariant object representation on several examples. First we illustrated the stability of the shape representation in terms of unary relations both visually and numerically. We then experimentally demonstrated the invariance of binary relations on a star-like object. We showed experimentally that the binary relations derived are invariant. The final example demonstrated the proposed approach as a tool for 3D object recognition. The aim was to recognise 3D objects in terms of planar faces. A hexagonal model shape was hypothesised in the image. The only instance of the hypothesised model was successfully recovered.

Acknowledgements

The work was partially supported by ESPRIT Projects RETINA and SAM.
3.5. CONCLUSIONS

Stability/Robustness of Invariants

Figure 3.1: Measurement Stability. Sub-figures arranged from top to bottom, left to right: 3.1.1 has basis size 3 pixels, 3.1.2 has basis size 6 pixels, 3.1.3 has basis size 9 pixels, 3.1.4 has basis size 12 pixels, 3.1.5 has basis size 15 pixels, 3.1.6 has basis size 18 pixels, 3.1.7 has basis size 21 pixels, 3.1.8 has basis size 24 pixels, 3.1.9 has basis size 27 pixels. Note that basis size is measured by area-perimeter ratio, and the basis used in the experiment is an equilateral triangle.
3.5. CONCLUSIONS

Figure 3.2: Model of "5": reference points are (43, 11), (20, 11), (10, 60)

Figure 3.3: Projection of "5": reference points are (71, 45), (41, 63), (95, 117)

Figure 3.4: Normalised model "5" in the affine coordinate system defined by its reference points

Figure 3.5: Normalised projection of "5" in the affine coordinate system defined by its reference points
3.5. CONCLUSIONS

Figure 3.6: Model of “F”: reference points are (11, 10), (54, 21), (11, 65)

Figure 3.7: Projection of “F”: reference points are (24, 102), (72, 50), (104, 125)

Figure 3.8: Normalised model “F” in the affine coordinate system defined by its reference points

Figure 3.9: Normalised projection of “F” in the affine coordinate system defined by its reference points

Figure 3.10: Model of “e”: reference points are (11, 10), (54, 21), (11, 65)

Figure 3.11: Projection of “e”: reference points are (24, 102), (72, 50), (104, 125)
3.5. CONCLUSIONS

Figure 3.12: Normalised model "e" in the affine coordinate system defined by its reference points

Figure 3.13: Normalised projection of "e" in the affine coordinate system defined by its reference points

Figure 3.14: Planar Projective Transforms
3.5. CONCLUSIONS

Figure 3.15: Example contour used for computing the invariant

Figure 3.16: Translation Invariants
3.5. CONCLUSIONS

Figure 3.17: Similarity Invariants

Figure 3.18: Affine Invariants
3.5. CONCLUSIONS

Figure 3.19: Perspective Invariants

Figure 3.20: Model of "star" for testing our binary measure: reference points are labelled with \( i = 1 \ldots 10 \)

Figure 3.21: Projection of "star" for testing our binary measure: reference points correspond to those in the model
3.5. CONCLUSIONS

Figure 3.22: Experiment 1: Matching a hexagon under perspective transform
Bibliography


Chapter 4

Colour Model Based Region Segmentation

4.1 Introduction

As described in Chapter 1, our strategy for recognising a 3D object is to recognise the individual surface patches first and then reconstruct the 3D object pose from the identified surfaces. We use patterns on these surface patches as the primary evidence for identification. In Chapter 2 we have introduced methods of extracting salient points from the edgel patterns on the surface patches. The unary and binary measurements in the relational representations described in Chapter 3 may be generated based on point features detected by the algorithms introduced in Chapter 2. In this and the following chapter, we describe an alternative way of producing relational representation of surface patterns. The primary features used here are regions.

The use of regions as primitives to construct high level image representations has many advantages over the use of edges.

1. Typically there are fewer regions than edge pixels. After segmentation the number of regions is often small enough to allow the construction of high level representations directly from them. As the number of edge pixels is often very large after edge detection, addition perceptual grouping by means of hough transform, polygonal approximation or curve fitting, or feature detection (such as that introduced in Chapter 2) are needed.
4.1. INTRODUCTION

These additional segmentation and feature detection processes introduce more parameters and thresholds to the whole vision system and make it difficult to control.

2. A small number of regions allows a more compact high level representation. This often leads to a significant speed improvement for the high level correspondence search algorithms.

3. Regions often have more meaningful semantic contents (such as colour, texture) than edge pixels. Such semantic contents provide important discriminant measures that can help in the high level matching process by indexing the candidate object labels and reducing the chance of generating false positives [3].

4. In terms of feature detection, regions are more stable and noise resistant than those primitives that depend on gray-level discontinuities. We will demonstrate this in the experimental section of this chapter. Later in Chapter 5, we show that the resulting affine invariant representations based on regions are also robust.

Colour is a very important cue in vision. It is a more stable attribute than mere luminance. Luminance (intensity) tends to change with a changing lighting condition, viewing angle, surface curvature and shadows, while the perception of colour is largely unaffected. This chapter focuses on the problem of extracting regions using colour models. In the next section, we discuss two issues relating to colour features, namely the issue of generating and matching of colours for recognition purposes and the issue of using jointly intensity and chromaticity information to achieve a more coherent segmentation.

In [3], Haralick has given a set of constraints for good region segmentation. In this chapter, we will analyse these constraints and their manifestations in different segmentation algorithms. We show that both global feature based clustering and local feature based region growing/watershed methods do not give satisfactory segmentations because some of these segmentation criteria were overlooked or not properly incorporated in the segmentation processes. A staged decision process and early thresholding makes these algorithms prone to error. The success of relaxation labelling lies in the ability to delay the decision making while integrating a more comprehensive set of constraints into a soft reasoning framework. Later in the chapter, we propose a piecewise-smooth model which enables us to integrate all the constraints of segmentation into a very simple label update formula. In the experiments,
we show that our regularisation-like algorithm performs better than the original Rosenfeld, Hammel and Zuckers relaxation labelling [4] [1] on the segmentation problem. Furthermore, it is able to express the degree of commitment in the final label probabilities.

In section 3, we introduce a set of constraints related to image segmentation. Some conventional segmentation methods are also compared here. Finally we introduce our own segmentation algorithm and compare it with other approaches.

4.2 Colour features for image segmentation

4.2.1 Colour coordinate systems

Here we are interested in segmenting (pulling out) the regions in an image that belong to the surface patterns of interest. In the model domain, we define a region to be a connected area on a single surface patch of the object. In the area, all points have a common surface reflectance. Assuming opaque objects, reflectance \( \rho(\lambda) \) is an invariant function where \( \lambda \) is the wavelength. Note that this is the physical definition of a region that is intrinsic to the object under investigation since the surface reflectance of the object does not change under lightening condition or projection. However, we can rarely measure the surface reflectance directly. What reaches our photo receptors is the reflected light,

\[
I(\lambda) = \rho(\lambda)L(\lambda)
\]

where \( L(\lambda) \) is the spectral power distribution of the light illuminating the object. This makes it difficult to resolve \( \rho(\lambda) \) from the incoming light alone unless we can measure or validly assume a certain model of \( L(\lambda) \). The problem becomes even more difficult when the spectral components of the reflected light is sensed in the form of responses of only a few (typically 3) types of photo receptors with different spectral sensitivity,

\[
\alpha_i(I) = \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} S_i(\lambda)I(\lambda)d\lambda \quad i = 1 \ldots N
\]

where \( N \) is the number of different types sensors, \( S_i(\lambda) \) is the spectral sensitivity of the \( i \)th sensor and \( \alpha_i(I) \) is the response of the \( i \)th sensor. Note that what is perceived by the later stage of the visual system (biological or artificial) is not the spectral lumination \( I(\lambda) \) but
4.2. COLOUR FEATURES FOR IMAGE SEGMENTATION

the colour response $\alpha_i(t)$. The convolution integral in 4.2 makes it impossible to resolve the spectral characteristics of the incident light from colour alone. Therefore measuring surface reflectance from normal photo receptors is an ill conditioned problem, and we can not guarantee a physical segmentation of regions with colour information alone. Here we make two assumptions to make the problem workable. First we calibrate our camera by imaging the scene with the same reference white source as that used for model creation, and we assume that the spectral characteristics of the light illuminating the object in the scene is the same as that in the model. Secondly, we relax the requirement of physical segmentation, and change the definition of a region into a connected area with all points having the same colour. Here colour refers to the response of our photo receptors, which includes both chrominance and luminance information. When the input image is processed by a Colour Constancy Algorithm, such colour based segmentation is possible even under a slightly changed illumination conditions.

Following the tri-chromatic theory of colour perception, most of the colours can be perfectly reproduced using a mixture (linear combination) of three fixed independent colours

$$C = C_1\alpha_1 + C_2\alpha_2 + C_3\alpha_3$$

Most of the artificial colour vision systems use this three dimensional representation. The choice of the three primaries and the geometry of the visible colour space jointly define the colour coordinate system. Each point in this 3D coordinate system corresponds uniquely to one colour. CIE RGB Spectral Primary System defines the colour space based on linear combination of three primaries at $R = 700.0nm$, $G = 546.1nm$ and $B = 435.8nm$. However not all visible colours are reproducible from the RGB system. This made CIE to specify the XYZ system, which is based on three hypothetical primaries. It yields a full set of visible colours. The NTSC television system uses a different primary system in order to minimise the transmission bandwidth. All these coordinate systems are linearly related to each other.

---

1. This is not always true when the object is placed under a different light source than that were used for model acquisition. Even with the same light source, regions in the shadow may receive different kind of illumination from regions illuminated directly by the light source due to inter-reflection. However such problems are extremely complicated to analyse. Here we avoid it by assuming surfaces under investigation are subject to rather weak inter-reflection which can be considered as small perturbations of the incident light.

2. Three colours are independent from each other if one can not produce the third as a linear combination of the other two.
4.2. COLOUR FEATURES FOR IMAGE SEGMENTATION

However according to psychophysical experiments, none of these systems is perceptually uniform. Colour distances measured in these coordinate systems are different from that of human eye perception. This motivated the definition of a number of perceptually uniform colour spaces such as \( L - a - b \) and \( U^* - V^* - W^* \) systems which rely on non-linear coordinate transformations. Besides perceptual uniformity, another important issue that is often considered when choosing a colour coordinate system is the separation of intensity information from chromaticities (i.e. normalisation). This is based on the fact that human colour perception can be best described in terms of hue, saturation and intensity, and the perception of hue and saturation is largely unchanged over a wide range of intensity. This is an important point for algorithms which use different points of view of the same scene (for instance stereo vision), and for the segmentation algorithms which need to handle curved surfaces and shadows [6].

For our segmentation problem, we argue that perceptual uniformity is not necessary. This is because we are interested in establishing correspondences between the models and the scene. As long as the colour coordinate system enables us to compute measurable differences and the colours of model and scene are measured from the same coordinate system, it will be a useable cue for building the correspondences. The machine perception of colour does not have to be the same as that of human. Perceptual uniformity is not necessary unless one wants to compare the results of segmentation by human eye with those given by computer vision algorithms [1].

The effect of shadows and different illumination/viewing angles can be cancelled out if we use two out of the three chromaticities which are calculated in the following way [7]

\[
c_i = \frac{C_i}{C_1 + C_2 + C_3} \quad i = 1, 2, 3
\]  

(4.4)

However, such normalisation might yield noisy and unreliable measurements when the surfaces are too dark (i.e. \( I = C_1 + C_2 + C_3 \approx 0 \)). Therefore we should avoid trying to segment a region using colour when it is very dark. It has been shown that human vision system uses different types of photo receptive cells for bright coloured vision and dark uncoloured vision. Some researchers also suggested vision system designs that can switch to black and white analysis for low intensity images while using colour analysis on well illuminated areas or scenes.
4.3. WHAT IS A GOOD SEGMENTATION?

4.2.2 Combining intensity and chromaticity

Segmentation using chromaticity information has the advantage that it is largely independent of image intensity variation. However, because intensity information is completely lost after normalisation like in 4.4, the segmentation algorithm will not be able to capture obvious region boundaries such as those between black and white. Surface patterns often manifest themselves through both colour and intensity information. We shall now discuss how to use intensity cue to further enhance the segmentation result.

Using intensity cue in segmenting colour images may seem to be contradictory to the need of segmentation under various intensities. Generally, if intensity information is taken into account in the segmentation process, the algorithm will be able to distinguish between black and white, but it will also split the continuous colour patterns at intensity changes. Some of such intensity changes are intrinsic to surface patterns, some are caused by shading (such as in the cylinder example in [9]) or shadow. Unfortunately, at lower level of the vision system, a segmentation algorithm based on both colour and intensity cues cannot distinguish the different causes of intensity changes.

However, the situation is slightly different when we are dealing with pattern mapped polyhedral objects. Since the surface patches are planar, the regions on these surface patches are also planar. Since the illumination can be expected to be uniform across a single surface patch, the segmentation can be done based on both colour and intensity information within the surface patch. Because of the shading characteristics of polyhedral objects, intensity variations due to shading only happen at the surface boundaries, and the algorithm will split the regions if variation is significant. By definition, our regions are planar, and therefore splitting the regions with common chromaticities at surface boundaries is desirable.

4.3 What is a good segmentation?

4.3.1 Spatial and feature domain constraints

Since the models of the regions of interest are known a priori, the problem of “pulling out” these regions from the background is a supervised segmentation problem. Unlike clustering algorithms (which are unsupervised), we may exploit this prior knowledge about the object to achieve more coherent segmentation. The better we know about the object of interest, the
4.3. WHAT IS A GOOD SEGMENTATION?

more precise and rich prior information can be fed into the segmentation process.

Supervised segmentation in the spatial-feature domain is essentially a pixel classification problem. Classifiers of all kinds can be employed and supervised training is also possible [1]. A trained classifier is attractive because it avoids manual modelling of every object that one wants to segment, and simplifies the classifier design. The model is learnt and applied by the same machine without human intervention. However, at present, the classifier training is still a very difficult problem. Trainable classifiers typically require huge amount of training data, and a great deal of off-line training effort. The number of pattern classes that the approach is capable of handling is still very small. Therefore manual modelling is still commonly used.

In this chapter we are mainly concerned with exploiting colour information in segmenting regions. Segmentation, as a low level vision process, often does not exploit the global structural knowledge about the scene. Hence decision making at this level can only be based on local measurements such as pixel colour and pixel level neighbourhood interactions. As local measurements are highly susceptible to noise, several global constraints have been explicitly or implicitly employed in many segmentation algorithms, including both supervised and unsupervised methods.

What is a good segmentation? In [3], Haralick informally described several constraints that a good segmentation has to satisfy.

1. Regions of an image segmentation should be uniform and homogeneous with respect to some characteristic such as grey level, colour or texture.

2. Adjacent regions of segmentation should have significantly different values with respect to the characteristic according to which they are uniform.

3. Region interiors should be simple and without many holes.

4. Boundaries of each segment should be simple, not ragged and spatially accurate.

5. A region has to be a connected group of pixels.  

\[ ^3 \text{This constraint was implicitly assumed and not stated in Haralick's description. We put it here because we think this is one of the most important constraint and is also independent from others.} \]
4.3. WHAT IS A GOOD SEGMENTATION?

Conditions 1 and 2 express the feature domain characterisation of region interior and region boundaries. Condition 4 is a pure spatial constraint for region boundaries. Conditions 3 and 5 are pure spatial constraints for region interiors.

4.3.2 Constraint satisfaction in conventional segmentation algorithms

To date, there are three kinds of algorithms that have been used for segmenting colour images. Most of them are mere generalisations of segmentation algorithms for monochromatic images. The first type of algorithms is typically based on split-and-merge [10], region growing [7] and watershed methods [11]. These algorithms put emphasis on the local feature and geometric relations (as expressed in condition 2,3,5) and generally perform well if the image is not too noisy. Another category of algorithms includes various kinds of clustering [13] and histogramming [12] [14]. Although relatively robust in the feature domain, these global methods typically suffer from the loss of the important spatial knowledge content caused by basing the segmentation process on histograms or clusters and from the dependence on thresholds [6]. The third class of methods exploits the local interaction models and employs Markovian Random Field models with the Bayesian decision theory. These methods are recent developments in colour image segmentation, and seem to be most flexible in handling feature space and geometric constraints [15] [16] [17].

Previously we have introduced five constraints for segmentation. Different algorithms handle these constraints differently. In the global methods, one would normally try to apply clustering or histogramming in the feature domain and by doing so, Constraint 1 would be satisfied. After an initial labelling, the geometric constraints like simple regions (Constraint 3) and connectivity (Constraint 5) are enforced by morphological processing and connected component labelling. Clustering and histogramming are typically dependent on thresholds. The type and extent of post processing by mathematical morphology is also difficult to specify. Both thresholding and ad hoc post processing may have a decisive impact on the segmentation result. This has made these global methods difficult to use and prone to error. Some efforts have been made to improve these global methods by integrating the spatial measurements into the clustering process, such as the use of high dimensional spatial-feature space combined representations. Higher dimensional methods are typically very slow. Moreover, the spatial distributions of the cluster members are often unpredictable and therefore diffi-
cult to model. It seems that spatial domain data demands a different cluster distance criterion (which is often like nearest neighbour) from that of the feature domain (which is often like the distance from the cluster mean). To design a clustering algorithm that works on the spatial-feature space domain, one needs a way of combining the two distance measurements effectively. Note also that Constraints 2 and 4 are not incorporated in this type of segmentation algorithms at all.

In the local methods such as region growing and watershed, the local geometric and feature space constraints such as connectivity (Constraint 5) and significant boundary signature (Constraint 2) are automatically enforced when one searches for points showing common feature domain properties in the immediate neighbourhood of the current pixel. The weakness of this type of algorithms is that global feature uniformity (Constraint 1) and region simplicity (Constraint 3) can not be enforced. Since only local neighbours and their measurements are considered each time, the feature measurements may drift away across a large geometric distance. Premature decisions on pixel labels irrespective of a larger context will make the process prone to error. Also, these methods require initial guesses about the positions of the regions of interest to seed the search process. Sometime, repeated random trials are necessary in order to get a correct result.

Methods based on Relaxation Labelling and Markov Random Field model seem to combine the bests of both global and local methods. Now we describe these methods in detail.

4.4 Region segmentation by relaxation operations

The use of relaxation labelling in image segmentation has become popular recently. Relaxation has been applied to segment grey level [18] [19] [20] [21] [22], colour [15] and texture images [23] [24]. The methodology involved in these algorithms ranges from stochastic relaxation [15] [21], discrete relaxation [18] [19] to probabilistic relaxation [23] [24]. The power of relaxation based segmentation lies in the ease of integrating a precise prior region model and spatial relational constraints into the segmentation process. The fact that it is not reliant on thresholds until the final stage of processing is also an attractive property of the approach. Since all constraints are enforced gradually and the decision making is always "soft", the performance of relaxation based segmentation algorithms degrades gracefully as
the input becomes more and more noisy. In this section we present a probabilistic relaxation based colour image segmentation algorithm. Unlike previous algorithms, our design utilises both intensity and chromaticity information in the segmentation. It is simpler and faster than the stochastic methods used by Daily [15], and is particularly suitable for the tasks of segmenting planar coloured regions. First let us look at two existing relaxation based segmentation algorithms in section 4.4.1 and 4.4.2, then in 4.4.3 we introduce our new segmentation algorithm.

4.4.1 Segmentation using KH Probabilistic Relaxation

Our approach is model based with the colour model of the $i$th region described in terms chromaticity distribution $P((x,y) \rightarrow i|c(x,y))$ where $c(x,y)$ is the 2D chromaticity vector (c.f. equation 4.4)

$$c = (c_1, c_2) = \left( \frac{C_1}{C_1 + C_2 + C_3}, \frac{C_2}{C_1 + C_2 + C_3} \right)$$

Given a point $(x,y)$ with chromaticity $c(x,y)$, $P((x,y) \rightarrow i|c(x,y))$ represents the probability of point $(x,y)$ belonging to the $i$th region independent of contextual information, i.e. $P((x,y) \rightarrow i|c(x,y))$ is the label probability based on the unary measure $c(x,y)$ alone.

Let us denote the probability of assigning label $i$ to point $(x,y)$ by $P((x,y) \rightarrow i)$. Note that the labels of the neighbouring pixels interact with each other due to the global uniformity constraint. The assumption is that all pixels in a region should have similar chromaticities and intensities (c.f. section 4.1). Accordingly it is reasonable to specify the label interaction in terms of the transitional probability. Let $(x_a,y_a)$ and $(x_b,y_b)$ be the neighbouring pixels. Then we have

$$P((x_a,y_a) \rightarrow i|(x_b,y_b) \rightarrow j) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma_c} \exp\left(-\frac{|c(x_a,y_a) - c(x_b,y_b)|^2}{2\sigma_c^2} - \frac{(I(x_a,y_a) - I(x_b,y_b))^2}{2\sigma_I^2}\right) & \forall j = i \\ \hat{P}((x_a,y_a) \rightarrow i) & \forall j \neq i \end{cases}$$

(4.6)

where $I(x,y)$ is the intensity component at point $(x,y)$. $\sigma_c$ and $\sigma_I$ control the variance of chromaticity and intensity components respectively. $\hat{P}((x_a,y_a) \rightarrow i)$ is the prior distribution of the label and is initialised with a predefined constant. Equation 4.6 states that when $i = j$, $(x_a,y_a)$ and $(x_b,y_b)$ are assumed to belong to the same region and therefore their colour and intensity should be similar. If $(x_a,y_a)$ and $(x_b,y_b)$ are not in the same region
4.4. REGION SEGMENTATION BY RELAXATION OPERATIONS

(i \neq j), then the two label assignments are independent from each other, and therefore
\[ P((x_a, y_a) \rightarrow i|(x_b, y_b) \rightarrow j) = \hat{P}((x_a, y_a) \rightarrow i), \]
which is the prior unconditional probability (often a predefined small constant).

Our goal is to compute the a posteriori probability of label assignment

\[ P((x_a, y_a) \rightarrow i|(x_b, y_b) \rightarrow j, \forall j, \forall (x_b, y_b) \in N_a) \quad (4.7) \]

where \( N_a \) is the set of pixels constituting the neighbourhood of point \((x_a, y_a)\). This a posteriori probability can be expanded using the standard Bayesian inference rule and total probability theorem. Assuming the conditional independence of the measurements, one can factorise the joint prior \( P((x_a, y_a) \rightarrow i, (x_b, y_b) \rightarrow j, \forall j \in \Omega, \forall (x_b, y_b) \in N_a) \) and obtain a label update rule [5]:

\[ P((x_a, y_a) \rightarrow i|(x_b, y_b) \rightarrow j, \forall j, \forall (x_b, y_b) \in N_a) = \frac{P((x_a, y_a) \rightarrow i)Q((x_a, y_a) \rightarrow i)}{\sum_j P((x_b, y_b) \rightarrow j)Q((x_b, y_b) \rightarrow j)} \quad (4.8) \]

where \( Q((x_a, y_a) \rightarrow i) \) is the support of \((x_a, y_a) \rightarrow i\) from \((x_a, y_a)\)'s neighbours. The support function in the above formula can be shown to be:

\[ Q((x_a, y_a) \rightarrow i) = \prod_{(x_a, y_b) \in N_a} \left\{ \sum_j P((x_a, y_a) \rightarrow i|(x_b, y_b) \rightarrow j)P((x_b, y_b) \rightarrow j) / \hat{P}((x_a, y_a) \rightarrow i) \right\} \quad (4.9) \]

Equations 4.8 and 4.9 are of the form of the original probabilistic relaxation formulated by Kittler and Hancock (KH PR) [5]. Note that the contextual information will be relatively low in our colour segmentation problem if we assume the unary colour measurement of the pixel to be more important in determining its label than the contextual constraint. In this case, we can approximate the transitional probability \( P((x_a, y_a) \rightarrow i|(x_b, y_b) \rightarrow j) \) with the prior \( \hat{P}((x_a, y_a) \rightarrow i) \) in some parts of the formula, and this results in a further simplification of the support function:

\[ Q((x_a, y_a) \rightarrow i) = 1 + \sum_{(x_a, y_b) \in N_a} \sum_j P((x_b, y_b) \rightarrow j)P((x_a, y_a) \rightarrow i|(x_b, y_b) \rightarrow j) - \hat{P}((x_a, y_a) \rightarrow i) / \hat{P}((x_a, y_a) \rightarrow i) \quad (4.10) \]
4.4. REGION SEGMENTATION BY RELAXATION OPERATIONS

Substituting \(4.6\) into \(4.10\), we obtain for the support function

\[
Q((x_a, y_a) \rightarrow i) = 1 + \sum_{(x_a, y_a) \in N_a} \left( \frac{P((x_a, y_a) \rightarrow i | (x_b, y_b) \rightarrow i)}{P((x_a, y_a) \rightarrow i)} - 1 \right) P((x_b, y_b) \rightarrow i)
\]

(4.11)

Support function \(4.11\) is the same as that suggested by Rosenfeld, Hummel and Zucker [4] and was successfully used for multi-spectral pixel classification by Eklundh [1].

The use of support function \(Q((x_a, y_a) \rightarrow i)\) with relaxation formula \(4.8\) allows the influence to a single pixel to be gradually extended to a quite large neighbourhood. This is the reason why relaxation labelling performs better than non-iterative local methods like region growing and watershed in terms of global feature uniformity constraint (Constraint 1). The contextual dependent decision making also makes relaxation labelling more robust to local error, and the resulting regions often do not contain many holes (Constraint 3). However as there is no label for the edge pixels, all pixels are either described in the interior of the region of interest or in the background. The failure to model the region boundaries and the overlooking of Constraints 2 and 4 can result in undesirable behaviour on the region boundaries when using the relaxation algorithm described in this section.

Consider a two class case where there is only one object label (described by probability \(P(x, y)\)) and one background label (described by probability \(1 - P(x, y)\)). The label update formula \(4.8\) can be written as:

\[
P((x_a, y_a) \rightarrow i | (x_b, y_b) \rightarrow j, \forall (x_a, y_a) \in N_a) = \frac{P((x_a, y_a) \rightarrow i) Q((x_a, y_a) \rightarrow i)}{\sum_j Q((x_b, y_b) \rightarrow j)}
\]

(4.12)

If the contextual support of the conjunctive form (like in \(4.9\)) is used, all object pixel labels in the neighbourhood are required to be positive \((\approx 1)\) in order to make a significantly large support. On a region boundary, however, not all of the neighbour pixels have object label with the probability close to unity. This will make \(P(x_a, y_a) \approx 0\) at the boundary pixels. In the next iteration, the object label at the pixel next to \((x_a, y_a)\) will also get penalised because \(P(x_a, y_a)\) becomes too small. Thus as the iterations proceed, the pixels appears to be eroding into the region. If the contextual support of the disjunctive form (like in \(4.10\)) is used, only a few object pixel labels in the neighbourhood are required to be positive \((\approx 1)\) in order to make a significantly large support. This will eliminate the erosion effect but will introduce dilation of the region if the binary conditional probability is biased towards giving a posi-
4.4. REGION SEGMENTATION BY RELAXATION OPERATIONS

A delicate balance has to be maintained by specifying the support function and the binary conditional probability in order to achieve the convergence at the correct region boundary. In [1], Eklundh got round the problem by obtaining the binary compatibilities from supervised training.

One may argue that such erosion/dilation problems are the result of a relaxation process that fails to take important unary measurements into consideration during the iterative label update process. Such an overlook of measurements can yield a result that drifts away from the initial observation. In [2], Stoddart proposed a new support formula of the following form:

\[ Q((x_a, y_a) \to i) = \prod_{(x_b, y_b) \in N_a} \sum_j P((x_a, y_a) \to i|(x_b, y_b) \to j)P((x_b, y_b) \to j|c(x_b, y_b))P((x_b, y_b) \to j) \]

(4.13)

Comparing 4.9 with 4.13, one can see that 4.13 does take unary measurement into consideration in every iteration of the labelling process. However, as one can see that the unary measurement probability \( P((x_b, y_b) \to j|c(x_b, y_b)) \) is integrated into the support function using a conjunctive operator: multiplication. This might not be correct for our segmentation problem, since the object label probability cannot be increased even when there is a positive unary evidence (i.e. \( P((x_b, y_b) \to j|c(x_b, y_b)) \approx 1 \)). The same erosion effect can still be expected to happen here.

4.4.2 Segmentation using Markov Random Field Model

In [15], Daily proposed a segmentation process as a simultaneous minimisation of four criteria functions. Now we briefly describe these criteria and their relation with the segmentation constraints introduced previously. The first criterion \( E_i \) minimises the colour differences between neighbouring pixels that belong to the same region:

\[ E_i = \lambda \sum_{x,y} ((f_{x,y+1} \Theta f_{x,y})^2(1-v_{x,y}) + (f_{x+1,y} \Theta f_{x,y})^2(1-h_{x,y})) \]

(4.14)

where vectors \( f_{x,y+1} \Theta f_{x,y} \) and \( f_{x+1,y} \Theta f_{x,y} \) represent the colour differences between point \((x, y)\) and its neighbouring pixels. \( v_{x,y} \) and \( h_{x,y} \) correspond to the vertical and horizontal line processes respectively and vary continuously between 0 and 1, with 0 representing the
absence of a line at pixel \((x, y)\) and 1 its presence. Minimising \(E_i\) will smooth out the noisy points within regions, and optimise the segmentation in terms of Constraint 1 and 3. The second criterion attempts to control fidelity between the resulting image \(f_{x,y}\) and the originally image \(d_{x,y}\) by minimising the squared distance:

\[
E_d = \alpha \sum_{x,y} |f_{x,y} - d_{x,y}|^2
\]  

(4.15)

When \(E_d\) criterion is used jointly with \(E_i\), we can see that Constraint 2 will be maximally satisfied. The third criterion introduces costs for inserting line pixels:

\[
E_l = \beta \sum_{x,y} (v_{x,y} + h_{x,y})
\]  

(4.16)

The fourth criterion forces the line processes to take on (1) or off (0) states and is defined as follows:

\[
E_g = \gamma \sum_{x,y} \left( \int_0^{v_{x,y}} g^{-1}(v) dv + \int_0^{h_{x,y}} g^{-1}(h) dh \right)
\]  

(4.17)

where \(g^{-1}()\) is standard sigmoid function representing the gain function for line processes. Optimising this criteria will ensure that the region boundaries are unambiguous. (i.e. are spatially accurate as described in Constraint 4). The global cost function \(E = E_i + E_d + E_l + E_g\), the binary line processes \(v_{x,y}\) and \(h_{x,y}\) plus a chosen lattice structure of the pixels forms a Hopfield network model of regions. The minimisation of \(E\) is achieved using a standard iterative update rule based on Mean Field Theory.

An important improvement of Daily’s method over the previously introduced Eklundh’s Relaxation Labelling method is the introduction of boundary labels (line processes). Criterion \(E_i\) excludes the boundary pixels from the smoothing process, thus avoids the excessive influence between pixel labels across the region boundary. Daily’s method was demonstrated on unsupervised segmentation (clustering) of the image data.

### 4.4.3 A new relaxation algorithm for image segmentation

In [15], Daily also introduced an important model for image segmentation which captures most of the constraints described by Haralick. Essentially, the model assumes that the image after a good segmentation is piece-wise smooth. Such piece-wise smoothness model (some-
time also called weak membrane model) has been widely used in many other regularisation problems such as contour, surface modelling, and noise suppression.

In Daily's formulation of the segmentation problem, this model captures Haralick's constraints nicely.

1. Within each region, the measurements are smooth with little variation. This captures the feature uniformity requirement in Constraint 1.

2. Between the regions, there are single step discontinuities. This corresponds to the requirement for spatially accurate boundary (Condition 4).

3. The membrane does not break at every discontinuity. There is a cost associated with breaking a membrane. The "gravity" of the system causes the membrane to break only at large significant feature discontinuities. This captures Constraints 2 and 3 nicely.

The above model suggests two alternative ways of obtaining a good segmentation. One way is to pre-process the image with a piece-wise smoothing regularisation process to reduce its noise and then segment it using any segmentation approaches introduced previously. We can expect the segmentation result obtained this way to be certainly good according to Haralick's constraints, irrespective of what type the original image is. Another way is to embed the piece-wise smooth model into the labelling process. In this way we can also expect the resulting labels to be consistent according to Haralick's constraints.

Now we introduce our method. First we introduce a simple relaxation labelling formula that embeds the piece-wise smoothness constraint. Later we describe a pre-processing filter that is suitable for denoising the image for segmentation purpose.

Let us consider a two class problem: one class is the region of interest, the other is the background. Consider a neighbourhood \((x_n, y_n), n \in [1 \ldots N]\) around pixel \((x, y)\). Denote the probability of \((x, y)\) being a region pixel with \(P((x, y) \rightarrow 1)\), and the probability of \((x, y)\) being a background pixel with \(P((x, y) \rightarrow 0)\). Assuming the support from each neighbouring pixel is dependent on the rest, the total number of possible outcomes for labelling the point \((x, y)\) is

\[
\sum_{n=1}^{N} \sum_{i=0}^{1} \sum_{j=0}^{1} P((x_n, y_n) \rightarrow j) P((x, y) \rightarrow i | (x_n, y_n) \rightarrow j)
\]

Since each region pixel has to be connected to at least one neighbouring region pixel, total
number of outcomes for \((x, y) \rightarrow 1\) supported by at least one of its neighbour is:

\[
\sum_{n=1}^{N} P((x_n, y_n) \rightarrow 1)P((x, y) \rightarrow 1|(x_n, y_n) \rightarrow 1) = (4.19)
\]

Hence we have the posterior probability formula:

\[
P((x, y) \rightarrow 1|(x_n, y_n), n \in [1 \ldots N]) = \frac{\sum_{n=1}^{N} P((x_n, y_n) \rightarrow 1)P((x, y) \rightarrow 1|(x_n, y_n) \rightarrow 1)}{\sum_{n=1}^{N} \sum_{i=0}^{1} \sum_{j=0}^{1} P((x_n, y_n) \rightarrow j)P((x, y) \rightarrow i|(x_n, y_n) \rightarrow j) = (4.20)}
\]

Define the conditional probability for label interaction:

\[
P((x, y) \rightarrow i|(x_n, y_n) \rightarrow j) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(c(x, y) - c(x_n, y_n))^2}{2\sigma^2}} - \frac{(l(x, y) - l(x_n, y_n))^2}{2\sigma^2} & \forall i = j \\ 0 & \forall i \neq j \end{cases} = (4.21)
\]

i.e. the smaller colour difference the greater the interaction between pixels of similar label.

Note that since \((x, y) \rightarrow 1\) and \((x, y) \rightarrow 0\) are mutually exclusive outcomes of one event, i.e. \(P((x_n, y_n) \rightarrow 0) = 1 - P((x_n, y_n) \rightarrow 1)\). From 4.21, we know that:

\[
P((x, y) \rightarrow 1|(x_n, y_n) \rightarrow 1) = P((x, y) \rightarrow 0|(x_n, y_n) \rightarrow 0) \neq 0 = (4.22)
\]

and

\[
P((x, y) \rightarrow 1|(x_n, y_n) \rightarrow 0) = P((x, y) \rightarrow 1|(x_n, y_n) \rightarrow 0) = 0 = (4.23)
\]

Substituting 4.22 and 4.23 into 4.20 and rearrange, we obtain the following label update formula:

\[
P^{k+1}((x, y) \rightarrow 1|(x_n, y_n), n \in [1 \ldots N]) = \frac{\sum_{n=1}^{N} P^k((x_n, y_n) \rightarrow 1)P((x, y) \rightarrow 1|(x_n, y_n) \rightarrow 1)}{\sum_{n=1}^{N} P((x, y) \rightarrow 1|(x_n, y_n) \rightarrow 1) = (4.24)}
\]

where \(k\) is the number of iteration. We argue that this label update formula performs better than the previous relaxation based segmentation algorithms without the problems of region dilation or erosion. This is because the neighbourhood connectivity defined in this formula best captures our notion of connectivity, i.e. as long as one neighbour is a region pixel, the pixel in the centre becomes connected to the region. Such definition handles the boundary...
situation well and is able to maintain fine scale structural features while eliminating the isolated peaks or holes which lack the contextual support. This labelling process should be initialised by the unary probability $P((x, y) \rightarrow 1|c(x, y))$.

The behaviour of this label update formula is similar to that of anisotropic gradient inverted iterative filtering which assumes a piece-wise smoothness model of the image. It anisotropically diffuses the measurements (here the probabilities) according to the binary relational evidences. Based on the same principle, we propose to use the following adaptive filter to regularise the image before segmentation according to the piece-wise smoothness assumption:

$$c^{k+1}(x, y) = \frac{\sum_{n=1}^{N} e^{n}(xy) + 1)P((x, y) \rightarrow 1|c(x, y) \rightarrow 1)\sum_{n=1}^{N} P((x, y) \rightarrow 1|c(x, y) \rightarrow 1)}{\sum_{n=1}^{N} P((x, y) \rightarrow 1|c(x, y) \rightarrow 1)} \quad (4.25)$$

where $P((x, y) \rightarrow 1|c(x, y) \rightarrow 1)$ is defined as in 4.21.

### 4.4.4 Binary and Unary Distributions

In order to recognise coloured planar patterns, we assume that each region forms a cluster in the chromaticity space. The distribution of the clusters can be measured by histogramming techniques during the model generation phase. The resulting cluster description can be stored in either parametric or histogram form, and is referenced by the segmentation algorithm through function $P(c(x, y)|(x, y) \rightarrow i)$.

The success of the algorithm depends on the accuracy of our unary and binary distribution model. Notice that in Equation 4.21 and 4.6 we have specified heuristically the binary measure distribution model as a Gaussian like function in terms of the chrominance distance $|c(x_a, y_a) - c(x_b, y_b)|$. However when the distribution are complex, a more precise approach is needed. One can measure the binary distribution through extensive experiments. Here we introduce an analytical solution to the problem.

Suppose we have two arbitrary points $(x_a, y_a)$ and $(x_b, y_b)$ belonging to the same region $i$. The chromaticity vectors $c(x_a, y_a)$ and $c(x_b, y_b)$ of the two points can be considered as two samples of two independent random processes drawn from an identical distribution $P(c(x, y)|(x, y) \rightarrow i)$. Note that the distance between the two chromaticity vectors $|c(x_a, y_a) - c(x_b, y_b)|$ is also a random process. Given the distribution of $c(x_a, y_a)$ and $c(x_b, y_b)$,
it is possible for us to find the distribution of pairwise distances between the random samples of two processes. In fact, the distribution of \( \delta c = |c(x_a, y_a) - c(x_b, y_b)| \) is the correlation between the distribution function of \( c(x_a, y_a) \) and that of \( c(x_b, y_b) \). In case that \( c(x_a, y_a) \) and \( c(x_b, y_b) \) follow the same distribution, the distance distribution is the autocorrelation of \( P(c(x, y)|(x, y) \rightarrow i) \)

\[
R_i = R_j = \int P(c(x, y) - c'(x, y) \rightarrow i)P(c(x, y)|(x, y) \rightarrow i)dc'
\]

(4.26)

Note that the intensity process is independent of colour histograms, and therefore it should remain unchanged. Hence the binary relational distribution should be

\[
P((x_a, y_a) \rightarrow i|(x_a, y_a) \rightarrow j) = \begin{cases} R_i & \forall j = i \\ 0 & \forall j \neq i \end{cases}
\]

(4.27)

4.5 Experiments

Experiments have been performed comparing the our segmentation algorithm with other algorithms.

It can be seen in Figure 4.1 that post-processing of the label probability map by morphology does not yield good results on noisy images where the initial classification error is high.

Relaxation with isotropic neighbourhoods (by KH-Probabilistic Relaxation) does a lot better. However as we can see from Figure 4.2, labelling probabilities tend to reduce from one iteration to another and the regions get slowly eroded. The reason for this is that an isotropic neighbourhood support function like 4.9 bases its decision on the conjunction of support form for all neighbours (c.f. Section 4.4.1).

The regularisation of probability map appears to produce the best result. This is because it has successfully integrated all 5 constraints, and most importantly the boundary constraints 2 and 4. It is this boundary constraint that prevents the labels to affect each other across region boundaries.

Finally Figures 4.5 and 4.6 show that even better segmentation can be achieved by using the proposed anisotropic filter to regularise the noisy image first, and then perform relaxation labelling.
4.6 Remarks and conclusion

In section 4.3, we described five constraints proposed by Haralick for region segmentation. These constraints are considered to be comprehensive in that they encompass both spatial and feature domain characterisation of a good segmentation. Specifically, Condition 1 and 2 in section 4.3 express the feature domain characterisation of region interiors and region boundaries respectively. Condition 4 is a pure spatial constraint on the region boundaries. Condition 3 and 5 are pure spatial domain constraint on region interiors.

We believe that a good segmentation process must be able to incorporate these constraints. We discussed the constraint satisfaction in the existing segmentation methods including spatio-feature domain clustering, region growing, water-shed, mathematical morphology and relaxation labelling. We argued that some of the problems appeared in these segmentation algorithms are often due to the fact that some of the constraints have been overlooked or not incorporated properly.

There are two principle ways of incorporating Haralick's constraints in a segmentation process:

1. One is to embed the constraints into an iterative labelling process. This has resulted Daily's Mean Field Theory based segmentation algorithm and our segmentation algo-
algorithm illustrated in subsection 4.4.3 (equation 4.24). We showed in the experiments that with better incorporation of Haralick's constraints, our segmentation algorithm has out performed the existing methods without suffering from their drawbacks.

2. Alternatively one can perform regularisation on the original image before segmentation, in such a way, the resulting processed image will suit Haralick's constraint better irrespective of the original image. The idea behind this approach is similar to that of feature detection theory introduced in Chapter 2, namely that, prior knowledge about the structure of the regions or features can be used to guide information filtering process in such a way that it will exclude irrelevant information and retain structures of interest. We showed in the experiments that the an-isotropic adaptive filter developed by us (equation 4.25) can be used effectively in pre-processing images to achieve better segmentation.
Figure 4.2: Relaxation with isotropic neighbourhood on noisy image: KH-Probabilistic Relaxation
(a) 1 iteration  
(b) 2 iterations  
(c) 3 iterations  
(d) 4 iterations

Figure 4.3: Segmentation with our relaxation algorithm on noisy image
Figure 4.4: Segmentation with our relaxation algorithm on clean image
Figure 4.5: Using anisotropic gradient inverted filter to regularise noisy image
Figure 4.6: Segmentation with our relaxation algorithm on noise filtered image: using anisotropic gradient inverted filter.
Bibliography


Chapter 5

Region Based Affine Invariant
Measurements: Unary and Binary Forms

5.1 Introduction

In this chapter we introduce methods for generating an affine invariant ARG representation of image and object models based on regions. Regions extracted using algorithms introduced in the last chapter may be used here.

In the introductory part of Chapter 3, we have discussed the need for a distributed affine invariant representation. Such a representation should have unary and binary measures that are invariant to affine transform, while enforcing the constraint that all objects in the graph undergo the same transformation. Such global constraint is commonly represented as a high order relation in conventional methods. Matching algorithms utilising high order relations are usually very slow. Chapter 2 has introduced a set of affine invariant measurements based on point features, from which one can construct an ARG representation of the scene using binary and unary relations only. This chapter introduces an invariant ARG representation based on region features. We start by looking at the problem of affine recovery from a mixture of point and moment features. Using recovered the affine parameters, we construct a unary measurement that is invariant to affine transform, and binary measures that enforce
5.2. AFFINE RECOVERY USING A COMBINATION OF POINT AND MOMENT FEATURES

We commence by introducing some basic measurements based on regions. Let \( R = f(x, y) \) be a region. Let us define the following:

- **Area**
  
  \[
  A = \int \int f(x, y) \, dx \, dy \quad (5.1)
  \]

- **Centroid**
  
  \[
  c_x = \frac{\int \int x \cdot f(x, y) \, dx \, dy}{A} \quad (5.2)
  
  c_y = \frac{\int \int y \cdot f(x, y) \, dx \, dy}{A} \quad (5.3)
  \]

- **2nd order moments**
  
  \[
  u_{2,0} = \int \int x^2 \cdot f(x, y) \, dx \, dy \quad (5.4)
  
  u_{0,2} = \int \int y^2 \cdot f(x, y) \, dx \, dy \quad (5.5)
  
  u_{1,1} = \int \int xy \cdot f(x, y) \, dx \, dy \quad (5.6)
  \]

Suppose region \( R \) is projected onto another image frame by a 2D affine transform. Let us represent the transform with 6 parameters \((a, b, c, d, t_x, t_y)\):

\[
\begin{align*}
  x' &= ax + by + t_x \\
  y' &= cx + dy + t_y
\end{align*}
\quad (5.7)
\]

The projected area of corresponding region \( R' \) in the second image frame will be

\[
A' = kA
\quad (5.8)
\]
where $k = ad - bc$. Similarly the projected 2nd order moments will be given by

$$u_{2,0}' = k \cdot \int \int (ax + by + tx)^2 \cdot f(x,y) \, dx \, dy = k(a^2 u_{2,0} + 2abu_{1,1} + b^2 u_{0,2} + At_x^2 + 2At_x(ac + bc))$$ (5.9)$$

$$u_{0,2}' = k \cdot \int \int (cx + dy + ty)^2 \cdot f(x,y) \, dx \, dy = k(c^2 u_{2,0} + 2cdw_{1,1} + d^2 w_{0,2} + At_y^2 + 2At_y(cc + dc))$$ (5.10)$$

$$u_{1,1}' = k \cdot \int \int (ax + by + tx)(cx + dy + ty) \cdot f(x,y) \, dx \, dy = k(acw_{2,0} + (ad + bc)u_{1,1} + bdw_{0,2} + A(txt + t_x(cc + dc) + ty(ac + bc)))$$ (5.11)$$

In the following we shall demonstrate how the affine transform between $R$ and $R'$ in the two image frames can be recovered.

5.2.2 Recovery of affine transform using mixture of point and moment features

Consider region model $R$ and its affine projection (scene) $R'$. We need to recover the 6 affine projection parameters from the given correspondence between $R$ and $R'$.

Traditionally, one could adopt two approaches to the affine recovery problem. One approach solves the recovery problem through the use of reference point features, such as corners, bi-tangents points, centroids etc. These methods often lead to a simple recovery formula, and it is also simple to generate invariant measurements from these reference point features. However, there are many disadvantages of using reference point features. In the introductory part of Chapter 4, we argued that there were four important reasons for region feature being preferable to point features. Moreover when we use reference points to construct projective invariant representations of models and scenes, additional problems may arise.

1. To construct a projective invariant representation, reference points are supposed to be invariant to the transformation. However the detection of projective invariant points
itself is a very challenging problem (as can be seen in Chapter 2). Most signatures of such points are affected by scale, shear and resolution, and the analysis and matching of these points under different resolutions and projective distortions can be quite difficult.

2. More importantly, since single reference points do not provide much useful information, several are often needed in order to construct a basis for invariant representation. Unfortunately, there is no guarantee that a sufficient number of feature points can be detected reliably for any arbitrary region under arbitrary projection.

3. Since more than one reference points are often needed to construct a basis, an additional perceptual grouping to partition the whole set of reference points into pairs, triples etc. need to be performed. Also one has to define a consistent ordering of the reference points in the pairs and triples. An exhaustive permutation of points is only applicable when the number of point is unrealistically small. Heuristics such as adjacency are often introduced to identify a good perceptual grouping. Additional parameters introduced in order to accomplish perceptual grouping has made the system more complex and prone to error.

An alternative approach to the affine recovery problem is to use global features instead of local point features. This category includes moment methods, 2D Fourier Transform etc. These methods typically take into account the information of every pixel in the region and therefore are more robust to noise and resolution change. They are also relatively easy to compute (since they do not involve feature localisation). However, these features are more sensitive to occlusions than localised point features. More importantly, transformation recovery formulas based on these measurements are often expressed in terms of simultaneous non-linear equations which can be difficult to solve analytically. Numerical search techniques are slow and susceptible to being trapped in local optima. Even when we can solve these equations, there could be many solutions due to the higher order terms in the formula. Thus additional measurements are needed to verify which one is valid. This verification process is expensive when there are many solutions, and more than one new measurements have to be introduced iteratively until we can pin-point the correct solution.

In this section we describe our recovery method which uses a mixture of point based
5.2. AFFINE RECOVERY USING A COMBINATION OF POINT AND MOMENT FEATURES

and region based measures (although all measurements have been obtained solely from region primitives). The algorithm combines the advantages of both local (point) feature based methods and global (moment) feature based methods while avoiding their disadvantages. It gives a single definite solution in an explicit form and therefore no verification or numerical equation solving is needed. It is much more robust to noise than edge / point feature based methods, and it is less vulnerable to the damage caused by occlusion than using moments alone. The method is generally applicable to many practical problems that require an affine recover or invariant representation, and is particularly suitable for constructing a distributed affine invariant representation of the scene from a collection of regions.

5.2.3 Normalising using two reference points

Suppose we are able to detect two points \((x_1, y_1), (x_2, y_2)\) invariantly with respect to region \(R\) in the model domain. Let us consider a problem of finding a projective transform that maps \((x_1, y_1)\) to \((1,0)\) and \((x_2, y_2)\) to \((0,0)\). The simplest projective transformation that achieves this is a similarity transform which is a special case of affine transform when \(a = d\) and \(b = -c\). Let this transform be \(T_n\)

\[
T_n = \begin{pmatrix}
a & -b & 0 \\
b & a & 0 \\
t_x & t_y & 1
\end{pmatrix}
\]

(5.12)

We can find out the parameters of this similarity transform by solving the following equation:

\[
0 = ax_1 + by_1 + t_x \quad 0 = -bx_1 + ay_1 + t_y
\]

(5.13)

\[
1 = ax_2 + by_2 + t_x \quad 0 = -bx_2 + ay_2 + t_y
\]

(5.14)

Now we have:

\[
T_n = \begin{pmatrix}
a & -b & 0 \\
b & a & 0 \\
t_x & t_y & 1
\end{pmatrix} = \frac{1}{k_n} \begin{pmatrix}
x_2 - x_1 & y_2 - y_1 & 0 \\
y_1 - y_2 & x_2 - x_1 & 0 \\
x_1^2 + y_1^2 - y_1 y_2 - x_1 x_2 & x_1 y_2 - y_1 x_2 & 1
\end{pmatrix}
\]

(5.15)

where \(k_n = (x_1 - x_2)^2 + (y_1 - y_2)^2\) which is the distance of the two reference points.
In the scene domain, suppose we can detect the affine projections of points \((x_1, y_1)\) and \((x_2, y_2)\) and let use denote them \((x_1', y_1')\), \((x_2', y_2')\) respectively. Again we can find the similarity transformation \(T_n'\) that maps \((x_1', y_1')\) to \((0,0)\) and \((x_2', y_2')\) to \((1,0)\):

\[
T_n' = \begin{pmatrix}
\alpha' & -b' & 0 \\
\beta' & \alpha' & 0 \\
t_x' & t_y' & 1
\end{pmatrix} = \frac{1}{k_n'} \begin{pmatrix}
x_2' - x_1' & y_2' - y_1' & 0 \\
y_1' - y_2' & x_2' - x_1' & 0 \\
x_1'^2 + y_1'^2 - y_1'y_2' - x_1'x_2' & x_1'y_2' - y_1'x_2' & 1
\end{pmatrix}
\tag{5.16}
\]

where \(k_n' = (x_1' - x_2')^2 + (y_1' - y_2')^2\).

The projected 2nd order moments of \(R\) and \(R'\) under \(T_n\) and \(T_n'\) respectively can be found using the standard affine moment projection formula which we have derived earlier.

### 5.2.4 Recovery of shear transform after normalisation

Mapping two feature points related to the region to \((0,0)\) and \((1,0)\) has significantly simplified the recovery problem. Substituting \((0,0)\) to \((x_1, y_1)\) and \((x_1', y_1')\) and \((1,0)\) to \((x_2, y_2)\) and \((x_2', y_2')\) into the affine projection formula

\[
x_1' = ax_1 + by_1 + t_x \\
y_1' = cx_1 + dy_1 + t_y
\tag{5.17}
\]

\[
x_2' = ax_2 + by_2 + t_x \\
x_2' = cx_2 + dy_2 + t_y
\tag{5.18}
\]

Solving the equations we have \(t_x = 0, t_y = 0, a = 1, c = 0\). These variables are removed from the affine transform and what is left is a representation for the shear component that remains after normalisation:

\[
T_s = \begin{pmatrix}
1 & 0 & 0 \\
b & d & 0 \\
0 & 0 & 1
\end{pmatrix}
\tag{5.19}
\]

Now we use the projection formula for area ratios \(k\) and 2nd order cross-moment that we have derived earlier to solve for \(b\) and \(d\). Suppose the similarity transform \(T_n\) has projected model region \(R\) to \(r\) and scene region \(R'\) to \(r'\), and let their area ratio be \(k = \text{area}(r')/\text{area}(r)\). Since the shear transform in 5.19 is a special case of affine transform, we apply the area ratio formula 5.8 to 5.19 and find \(d = k\). Substituting \(a = 1, c = 0, d = k, t_x = 0\) and \(t_y = 0\) into
5.2. AFFINE RECOVERY USING A COMBINATION OF POINT AND MOMENT FEATURES

5.2.1 We find \( b = (u'_{1,1} - k^2u_{1,1})/(k^2u_{0,2}) \), where \( u'_{1,1}, u_{1,1} \) and \( u_{0,2} \) are 2nd order moments of \( r' \) and \( r \). Now we have the shear component represented as:

\[
T_s = \begin{pmatrix}
1 & 0 & 0 \\
(u'_{1,1} - k^2u_{1,1})/(k^2u_{0,2}) & k & 0 \\
0 & 0 & 1
\end{pmatrix}
\]

5.2.5 Recovering the complete affine transform

The complete affine transform from region \( R \) to \( R' \) is

\[
T = T_n T_s T_n^{-1}
\]

In the above formula \( T_n \) represents a transformation that maps the model region \( R \) into a normalised coordinate system where the corresponding two feature points of the pair are situated at (0,0) and (1,0). \( T_s \) represents a shear-scaling transformation between the normalised model \( r \) and normalised scene \( r' \). \( T_n^{-1} \) serves as a mapping from the normalised coordinate system to the scene coordinate system.

The application potential of the recovery algorithm is enormous. The recovered affine transform can be used as a direct hypothesis in a hypothesis-verification recognition paradigm for recognising a set of regions. One can also use the recovered local affine transform to generate invariant binary measurements, and form an attributed relational graph representation of model and scene. The recovery formula can also be used directly for computing image motion and for recovering shape from texture. The next section demonstrates a stereo matching application.

5.2.6 Effect of noise and occlusion

It is well known that area and 2nd order moments are quite robust to noise since they involve integration over the entire region. High order moments have been criticised for being sensitive to occlusion. Abu-Mostafa & Psaltis [1] have shown that the signal to noise ratio of the moment measurement \( u_{pq} \) is proportional to \( 1/\sqrt{p+q} \) i.e. moments become more susceptible to noise as their order increases. Note that although moments themselves are unbiased measurements, their use in the recovery algorithm is not linear, and therefore may create
biased estimates of affine parameters. It is therefore always desirable to use moments of as low order as possible. Note that in our recovery formula, only one parameter $b$ is actually determined using 2nd order moment measures. The rest of the 5 affine parameters are all derived from area and reference points. Provided the reference points are extracted based on local measurements robust to noise, we can expect this formulation to be quite resilient to occlusion and noise.

5.2.7 Obtaining the reference points

Now we discuss how to obtain the two reference points.

Centroids are good candidates for reference points because they can be detected invariantly with respect to arbitrary affine transform. They are global features, and therefore the recovery that uses centroids directly or indirectly can be very robust against noise but not very robust against occlusion. However, we know that centroids are first order moments, and the amount of error due to occlusion will be much less severe as that for purely moment based methods which typically involve high order terms.

Moreover, we can group the regions into pairs, and use the joint moments, joint area and the two centroids to recover the affine transform. When the distance between the two centroids is large, occlusion is only likely to happen to one of the two regions, and the effect of occlusion on the position of the centroids will often be very small compared to the distance between the two centroids. In such cases, the system appears to be very robust to both noise and occlusion. In the experiments we observed that our centroid based pairwise affine recovery algorithm will produce less than 5% error in the parameters in the presence of up to 30% occlusion (c.f. Figure 5.17,5.18 in section 5.5). More importantly, the error performance of our algorithm degrades rather gracefully as noise and occlusion becomes severe.

Pairwise representation has one drawback: namely that it requires the regions to be grouped in pairs. If no additional cues are available (such as region adjacency etc.), we need to generate all possible pairs of regions. This may be computationally expensive for the matching algorithm, if the the number of the regions is large. In such cases, we will need to extract another invariant point apart from the centroid from a single region.

The choices are many. Any of the point features introduced in Chapter 2 may be used here as the second reference point. Points of maximum concavities, inflexion points or bi-tangent
points are good candidates of choice. Here we do not need to worry about the permutation order of using the reference points because the centroid of a region and a point on the contour have obvious geometric meanings. Since our recovery formula demands only two reference points and one (the centroid) is already available, a one dimensional search for the second reference point (along the region boundary) is fairly inexpensive. Moreover, when we select points with certain geometric meaning (such as bi-tangents or inflexion), the search space can be limited to only a few points. Often, the non-optimal candidates can be easily pruned out by fitting the projection formula of $u_{20}$ and $u_{02}$, which are not used in our recovery formula.

If one can obtain a third reference point, the affine recovery problem is trivially easy. The binary and unary invariant measurements can be directly generated from the method introduced in Chapter 3. However, our approach has important advantages over a purely point based approach. Our method is partially based on global features and is robust to noise. It does not require perceptual grouping of salient points into pairs or triples, and it does not depend on the permutation order of the reference points either. Also the detection of regions, and the calculation of low order moment features are both much more reliable and relatively insensitive to scale / resolution changes.

### 5.3 Fast stereo matching without feature detection

Note that our recovery formula applies to arbitrary regions containing arbitrary “grey shades”. Suppose we have a stereo pair $L(x, y)$ and $R(x, y)$, for a scene consisting of planar objects. We can assume the stereo disparity between $L(x, y)$ and $R(x, y)$ is due to an affine transform.

In order to use the affine recovery method introduced above, we need to obtain two reference points for each image. Alternatively, we can generate two regions for each image, and use the pair of centroids and the joint area, and moments. This can be done by splitting the colour images using spectral (or colour channel) filters. The simplest form of colour channel filter is to use the 2 chromaticity channels directly. Each channel forms an independent monochromatic image, and the two images formed by the two channels can be used as a region pair for affine recovery. One crucial criterion for choosing the channel filters is that the resulting image pairs have to have different centroids. The farther the centroids
of the sub-images from each other, the better the recovery method will work. This suggests that the two chromaticity channels should be as independent as possible. Since most of the images contain objects emitting/reflecting a fairly wide spectrum of light, the resulting monochromatic images from the two channels are often correlated with each other to a certain degree. It is therefore desirable to choose the two colour channels in such a way that the two portions of the visible spectrum occupied by them should be as far apart as possible and have a minimum overlap. This suggests that the preference of chromaticity channel pair such as \( r = \frac{R}{R+O+B}, b = \frac{B}{R+O+B} \) over channels pairs like \( r = \frac{R}{R+O+B}, g = \frac{O}{R+O+B} \) or \( g = \frac{O}{R+O+B}, b = \frac{R}{R+O+B} \). For grey-level images we can split the left/right images using intensity channel filters. For example, we can use a threshold to partition the intensity histogram into two parts and form two images based on each part of the histogram. It is also possible to match colour stereo pairs using combined chromaticity-intensity channel filters. If the image splitting process yield images with their centroids very close to each other, one would have to change the channel parameters and attempt the splitting again. Prior knowledge about the scene can certainly help in designing suitable channel filters.

Once the left and right images are successfully split into non-co-central filtered images, we can directly apply the above recovery formula to find out the affine transform between the left and right image. The greatest benefit of this stereo matching method is that it does not require any feature detection or feature matching and it therefore is very fast. Spectral-intensity based image splitting is a process that can be parallelised at pixel degree, and can be performed by many existing image acquisition hardwares.

5.4 Unary and binary measures

As in Chapter 3, affine invariant unary and binary measurements can be formed using basis matrices. Let \( B \) denote a basis matrix for region \( R \), and let \( P \in R \) be the homogeneous coordinate of an arbitrary point on \( R \). Given transformation \( P' = PA \) where \( A \) is a linear transform matrix, the barycentric coordinate \( C = PB^{-1} \) of point \( P \) can be shown to be invariant to affine transform provided basis matrix \( B \) is transformed as a vector. According to the group representation theory, measurement matrix \( B \) is transformed as a vector under linear transform \( A \) if and only if \( B' = BA \). Thus we can show that \( C' = P'(B')^{-1} = PA(BA)^{-1} = \)
5.4. UNARY AND BINARY MEASURES

\( PB^{-1} = C \) is invariant to affine transform. Therefore \( C \) can be used directly to compute affine invariant unary measures. If we have two region pairs \( R_1 \) and \( R_2 \) with basis matrices \( B_1 \) and \( B_2 \), then we can show that \( B_1B_2^{-1} \) form a binary measurement that embeds common affine transformation constraint (c.f. proof in section 3.3.1).

Now we show how to find such basis matrices from the affine recovery formula introduced in Section 5.2. Define a standard region \( R_0 \) with the two feature points situated at \((0,0)\) and \((1,0)\). Let the area of the region be 1, and its second order cross moment \( u'_{1,1} = 0 \). According to the recovery formula, for an arbitrary region \( R \), the transformation from \( R \) to \( R_0 \) is

\[
T = T_nT_sT_n^{-1}
\]  
(5.22)

where \( T_n \) is as calculated as in 5.16. Since the two point features in \( R_0 \) are already at \((0,0)\) and \((1,0)\), we have \( T_n^{-1} = I \). Substituting \( u'_{1,1} = 0 \) into 5.20, we have

\[
T_s = \begin{pmatrix}
1 & 0 & 0 \\
-u_{1,1}/u_{0,2} & k & 0 \\
0 & 0 & 1
\end{pmatrix}
\]  
(5.23)

We use \( B = T^{-1} \) as our invariant basis. Combining 5.23, 5.22, 5.15 and 5.16, we have

\[
B = k_n \begin{pmatrix}
1 & 0 & 0 \\
-u_{1,1}/u_{0,2} & k & 0 \\
0 & 0 & 1
\end{pmatrix}^{-1} \begin{pmatrix}
x_2 - x_1 & y_2 - y_1 & 0 \\
y_1 - y_2 & x_2 - x_1 & 0 \\
x_1^2 + y_1^2 - x_1y_2 - x_1x_2 & x_1y_2 - y_1x_2 & 1
\end{pmatrix}^{-1}
\]  
(5.24)

where \( k_n = (x_2 - x_1)^2 + (y_2 - y_1)^2 \). Therefore the barycentric coordinate of the arbitrary point \( P \) in the region is given by \( PB^{-1} = PT = PT_nT_s \). As before, to prove \( B \) is a valid basis, we need to show that it is transformed as a vector. Assuming region \( R' \) is the affine projection of \( R \) with transformation \( T_n \), we can obtain the required proof by expanding \( BT_n \) and showing that it is equal to \( B' \). The proof is somewhat lengthy but straightforward. The expansion of the matrices can be easily done by many symbolic computation packages (such as Maple and Reduce).
5.5 Experiments

This section provides an experimental verification for our proposed representation.

First we demonstrate that the basis given in 5.24 generates stable unary invariant measures. Figures 5.1 5.2 5.5 and 5.6 show a group of four objects ("club", "leaf", "bull", "plane") in four different kinds of configurations. Figures 5.3 5.4 5.7 and 5.8 show the unary measurements obtained for all individual objects in Configuration 1, 2, 3 and 4 respectively. From these Figures, one can see that the unary measurements of these objects do not change under arbitrary affine transform. For each object, a single reference point on the boundary as marked in 5.1 5.2 5.5 and 5.6 is extracted. This reference point together with the centroid, and the second order moments of the object allow us to compute basis $B$ using Equation 5.24. The invariant unary representation is calculated using formula $PB^{-1}$.

As described the Chapter 3, a binary measure between two objects can be considered as the relation between their bases. The simplest form of binary relation is $A_{ij} = B_j B_i^{-1}$. Note that basis matrices are always non-singular given that areas of the objects are not zero and the reference points do not overlap. For affine transform, binary relation $A_{ij}$ are 3x3 matrices, with last rows being constant vectors (0, 0, 1). Hence only six parameters are needed to define a binary relation under affine transform. These six parameters have distinct geometric meanings, namely that they describe the affine transform that maps basis $B_i$ to basis $B_j$. We can visualise the transform by using it to project a polygon defined by five points as illustrated in Figure 5.9. An affine projection of this shape would look like 5.10. Note that the position, orientation and shape of the projected polygon unambiguously define the underlying binary relation. Figures 5.11 and 5.12 show that Configuration 1 and Configuration 2 have an almost identical set of binary relations. Referring to Figures 5.1 and 5.2, we can see that this is because the four objects in the images undergo the same affine transformed. In 5.5 and 5.6, each object is transformed independently, therefore their binary measures shown in 5.13 and 5.14 are very different from those of Configuration 1. Note that in all cases, the unary measures of the objects do not change. Thus we have confirmed that we have an orthogonal decomposition of measurement into unary and binary relations and our binary relations precisely capture the global transformation constraint.

Figures 5.15 to 5.20 illustrate that affine recovery from pairs of regions are quite robust against both noise and occlusion. When the regions are well separated, occlusion shown in
Figure 5.17 has little effect on changing the two centroid and joint moments of the regions. Figure 5.18 shows that the recovered affine transform is still quite accurate. This is a property that is not exhibited by the pure moment based approach. Figures 5.19 and 5.20 shows that although the region contour and its interior can be heavily corrupted due to noise, the affine recovery using our formula is still quite reliable. This is a property that pure salient contour curvature feature based approaches do not have.

Figure 5.1: The group of four objects in Configuration 1

Figure 5.2: The group of four objects in Configuration 2

Figure 5.3: Unary measures obtained from Configuration 1 using basis given by 5.24

Figure 5.4: Unary measures obtained from Configuration 2 using basis given by 5.24
5.6 Remarks and conclusion

In this chapter, we have discussed how to construct ARG representations of model and scene based on region primitives under global affine transform. Like in Chapter 3, we use barycentric coordinates to generate our unary and binary measures. To obtain the barycentric coordinates we need to find a linear basis for each primitive region such that it satisfies the three conditions described in Section 3.3.1.

The problem of finding such linear affine invariant basis for a region is equivalent to the problem of recovering affine parameters given a region and its affine projected image. Existing affine recovery methods work either purely based on reference points or purely based on global regions features such as moments. Both approach have its advantage and disadvantage. Reference point based methods is relative robust to occlusion, but they need to form triplets and generate consistent ordering of points, this can results large number of hypothesis. While moment based methods do not need to form large number of hypothesis, they are less robust to occlusion and their recovery equations are more difficult to solve.

In this chapter, we proposed a new affine recovery formula that is based on a mixture of point and moment features. Our method combines the advantage of local (point) feature based methods and global (moment) based methods, while avoiding their disadvantages. It is more robust to noise than point based methods, and requires no feature localisation or grouping of primitives into consistent order. Compared to pure moment based or Fourier
domain methods, our method is much easier to use for it gives a single definite solution in explicit form, and it is also less sensitive to occlusion.

Distributed image representation like ARG is powerful in that it can handle incomplete and noisy data, it has not been use widely on high level matching problems due to the difficulty of handling global projection constraints in the representation. Upto this point we have demonstrated how to generate ARG representation of scene and model under affine transforms using both point and region primitives. Most importantly, these ARG representations are able to enforce global transformation constraint using binary and unary measures only. Thus enables the use of efficient deterministic matching algorithms to perform pattern matching.
5.6. REMARKS AND CONCLUSION

Figure 5.9: The five point polygon used to designate different binary relations. The vertices are (0,0), (1,0), (1,1), (0,1) and (0,−1).

Figure 5.10: An affine projection of the polygon

Figure 5.11: The binary relations between the objects in Configuration 1

Figure 5.12: The binary relations between the objects in Configuration 2
5.6. REMARKS AND CONCLUSION

Figure 5.13: The binary relations between the objects in Configuration 3

Figure 5.14: The binary relations between the objects in Configuration 4

Figure 5.15: A pair of model regions

Figure 5.16: An affine projection of the model regions
5.6. REMARKS AND CONCLUSION

Figure 5.17: An affine projection of model regions with occlusion

Figure 5.18: Back-projection of the model object using the recovered affine transform parameters from occluded scene

Figure 5.19: An affine projection of model regions with noise

Figure 5.20: Back-projection of the model object using the recovered affine transform parameters from noisy scene
Bibliography


Chapter 6

Contextual Dependent Labelling

6.1 Introduction

This chapter is concerned with using deterministic relaxation to solve matching problems in computer vision. Following the formulation in [1], we model the visual matching process as finding the correspondence between vertices of two attributed relational graphs (ARGs). Formally, we define ARG as a quadruple $G = (V, E, u_i, x_{ij})$ where $V = \{v_i | i \in [1, N]\}$ is a set of $N$ vertices (or nodes). $E \subset V \times V$ is a set of arcs (or edges), $u_i : V \rightarrow \mathbb{R}^n$ is the unary measurement function associated with each vertex $v_i$ and $x_{ij} : V \times V \rightarrow \mathbb{R}^m$ is the binary measurement associated with each pair of vertices $v_i$ and $v_j$. Label assignment function $L : V \times V' \rightarrow [0, 1]$ measures the degree of match between vertex $v \in V$ in graph $G$ and vertex $v' \in V'$ in graph $G'$. If we assume arbitrarily that $G$ represents the model and $G'$ represents the scene, pair $(v, v')$ is called an interpretation of the scene vertex $v'$ with model vertex $v$ and $L(v, v')$ represents the uncertainty of this interpretation with value 1 as positive certain, 0 as negative certain, 0.5 as maximally uncertain. Let $i \neq j \in [1, N]$, $(v_i, v')$ and $(v_j, v')$ be two different interpretations of scene $v'$. If both $L(v_i, v')$ and $L(v_j, v')$ are near 1, then we have multiple interpretations for $v'$. Sometime, we also say that the label of $v'$ is ambiguous. Given a collection of interpretations (a Joint Interpretation) $\alpha \subset \{(v_i, v'_p) | i \in [1, N], p \in [1, N']\}$, if for each $v'_p$ there is at most one interpretation, then we call $\alpha$ a Non-Ambiguous Joint Interpretation (NAJI). Further more, if for each model $v_i$ there is at most one object $v'_p$ corresponding to it, $\alpha$ is a One-to-One Joint Interpretation or Basic Joint Interpretation (BJI). It is clear that a BJI is equivalent to a one-to-one mapping (not function) between nodes in $V$ and nodes in $V'$. The
reason why it is called "basic" is that we can treat any collection of interpretations as a union (disjunction) of a group of Basic Joint Interpretations. We will explore the utility of this idea further in Section 3.

The goal of the labelling process is to find function $L$ that minimise a cost function that represents the total discrepancies between corresponding unary and binary measurements in the two graphs [2]. Searching for such function $L$ is generally an exponentially complex task. Due to the strong interaction between the label assignments, the optimisation surface is often highly irregular with many local minima. Therefore simple search strategies cannot give good results. Relaxation process has been introduced to alleviate the complexity. Label functions are calculated slowly and iteratively, and this allows the relaxation of global constraints, and representing them as collections of independent local constraints. This necessitates the small neighbourhood assumption in [5] and many other relaxation labelling algorithms. Fluctuations of the label functions can be introduced deliberately to improve the quality of the result. The resulting algorithms are called stochastic relaxation [3]. Deterministic relaxation updates the label function deterministically and there is no need to average the random fluctuation, hence they are generally faster than their stochastic counterparts. In this chapter, we will focus on deterministic relaxation algorithms. In particular, we assume that the certainty function of an interpretation $l_{ij} = L(v_i, v'_j)$ is entirely determined by the unary and binary measurements within the neighbourhoods of the two vertices $v_i$ and $v_j$, and therefore $l_{ij}$ can be updated independently. All relaxation algorithms discussed in this chapter will be object-centred [5].

Traditionally, the design of the update formulae for deterministic relaxation labelling is very heuristic and problem specific. Recently there have been several attempts to generalise the methodology using available mathematical frameworks, notably the probability and Bayesian theory [5] [1] and Fuzzy logic [4] [6] [9] [10] [8]. Among these algorithms, probabilistic relaxation [5] [1] has been considered to be a success both in terms of the theoretical soundness and the proven practical effectiveness. Some early algorithms have also been shown to be the derivative of probabilistic relaxation [1]. However, some fundamental problem (such as the problem of label amplification, dilution, multiple interpretation, design of null attractor etc) in the relaxation algorithms remain unsolved. Some of these problems are not unique to probabilistic relaxation but are common in many of the existing relaxation
6.2 Partition Function and Information Fusion Operator in Relaxation Labelling Algorithms

As the matching between two ARG can be inexact, we need a mathematical model of the uncertainty in the measurements as well as frameworks in which we can reason with uncertain information. Here we choose to use real number in $[0, 1]$ to represent uncertainty. Both probability theory and fuzzy set theory use this convention, and therefore formulation $L : V_1 \times V_2 \rightarrow [0, 1]$ has generality. It should be noted that any label function updating formula should yield values within range $[0, 1]$ so that the next stage of reasoning can use the results of previous ones successively or iteratively.

In order to construct an update formula, one needs to choose one or two or more information fusion operators. In [11] and [12], there is an excellent survey on available fusion operators and their properties. According to Bloch's [11] classification, there are three types of fusion operators:

- $F$ is conjunctive if $F(x, y) \leq \min(x, y)$
- $F$ is disjunctive if $F(x, y) \geq \max(x, y)$
- $F$ is compromise if $\min(x, y) \leq F(x, y) \leq \max(x, y)$

Probabilistic relaxation labelling as in [1] uses multiplication as conjunction operator and addition as disjunction operator. With no genuine compromise operator, disjunction(addition)
6.2. PARTITION FUNCTION AND INFORMATION FUSION OPERATOR IN RELAXATION LABELLING ALGORITHMS

is used assuming there is no correlation between the evidences. Using addition as disjunctive operator does not guarantee the result to be limited in \([0, 1]\), partition function is used to normalise the result:

\[
l_{ij} = \frac{Q_{ij}}{\sum_p Q_{pj}}
\]

where \(Q_{ij}\) is the support (including both unary and binary evidences), \(l_{ij}\) is the label function. The use of partition function has introduced some additional problems. First, it is based on the assumption that two conflicting interpretations of the same object are two mutually exclusive outcomes of one event, and the sum of the probabilities of all mutual exclusive outcomes of an event is 1. This assumption does not hold when there is need to find multiple interpretations of the same object. We call this the "multiple interpretation problem". Secondly, when the evidence gives a strong support to two or more equally valid interpretations of the same object, the sum-to-1 constraint tends to make the system swing to one of the interpretations or in the extreme cases, penalises all interpretations. The label values in this situation, appear to be diluted and no longer represent the amount of support they receive from the evidence. This is the problem of "label dilution". Thirdly, when the evidence suggests that there is no valid interpretation, the support values are all very small. But with the normalisation by partition function, the label values of all interpretations are amplified to make the sum to be 1. This is the problem of "label amplification". The traditional solution to label amplification is to add a null attractor. Those objects that do not have any valid interpretation are assigned to this null label. Thus every object will have one and only one interpretation. The problem with this null attractor is that it has to compete with other interpretations during the labelling process and it is often tricky to design an appropriate null attractor that is neither over competitive nor over modest. The difficulty is that the probability density functions of the features of the null attractor are unknown, and in theory, not independent of the probability distributions of features of the model. Thus extra effort to fine tune the null attractor is often needed when the type of model and/or the type of scene has changed.

For fuzzy logic we use \(\min(x, y)\) as a conjunctive operator and \(\max(x, y)\) as a disjunctive operator. Note that \(\max(x, y)\) is closed on the domain \([0, 1]\) therefore it does not need normalisation of any kind. One of the earliest fuzzy relaxation algorithms was proposed by
Rosenfeld in [4], which uses min(x, y) and max(x, y) exclusively:

\[ l_{ij}^{k+1} = \min_j \{ \max_{pq} \{ \min (f(x_{pi}, x_{qj}), l_{pq}^k) \} \} \] (6.2)

where \( f(x_{pi}, x_{qj}) \) is bounded in [0,1], and represents the degree of match between binary measures \( x_{pi} \) and \( x_{qj} \). The lack of compromise operator in the formulation makes the behaviour of the system strongly resemble boolean logic (where compromise does not exist). Ironically, compromise operator is one of the essential features for reasoning with uncertainty. Without it, any decision making is forced to be either over indulgent or over skeptical to collections of evidences. In the case of algorithm (6.2), it was criticised to be too skeptical (due to the outer-most min() operator) in the presence of a low support in the local area and tendency of driving the whole system to the negative certain state everywhere. Another fuzzy relaxation labelling algorithm was given by Ranganath [6]. It is essentially a combination of using fuzzy fusion operators min() and max() with summation and partition function. Therefore it also has all the problems introduced by the partition function.

In the next section we introduce a new relaxation labelling algorithm and its non-iterative variant. In the proposed algorithms (6.6) (6.12) we use a mixture of conjunction (min), disjunction (max) and compromise (average) operators to give us a good expressive power. Average is used as a genuine compromise operator, therefore eliminating the need of partition function. Multiple interpretations are inherently supported by the scheme.

6.3 A Relaxation Labelling Algorithm Based on Fuzzy Set Theory and Its Non-iterative Variant

First we review some basic definitions in fuzzy set theory in the context of the labelling problem. As in [13], a fuzzy set is described as a vector within the \( n \) dimensional unit hypercube \([0,1]^n\). Each dimension of the vector is a number within \([0,1]\) representing the degree of presence of a particular element that belongs to a universe of discourse \( U \) which is a non-fuzzy set with \( n \) elements.

Given two attributed relational graphs \( G = (V, E, u, x) \) and \( G' = (V', E', u', x') \), we need to find a label function \( l_{pq} = l(v_p, v'_q) \) for all \( v_p \in V, p \in [1,n] \) and \( v'_q \in V', q \in [1,n'] \). In
our formulation of the problem, the universe of discourse $\lambda = V \times V'$ is the collection of all possible labelling. Therefore the set of label values $L = \{l_{11}, l_{12}, \ldots, l_{pq}, \ldots, l_{nw}\}$ represents a fuzzy set on $\lambda$.

Consider label assignment from node $v_i \in V$ to node $v_j' \in V'$. Let the set of neighbouring nodes of $v_i$ be $U$ and the set of neighbouring nodes of $v_j'$ be $U'$. A Basic Joint Interpretation $\alpha$ of the neighbourhoods $U$ and $U'$ consists of a number of pairs $(p, q)$ where $v_p \in U$ and $v_q' \in U'$. We can view $F_{ij}^0(\alpha) = \{l_{pq} | (p, q) \in \alpha\} \subset L$ as a fuzzy set that contains our prior knowledge about the interpretations on neighbourhoods of $v_i$ and $v_j'$, and $F_{ij}^b(\alpha) = \{f(x_{pi}, x'_{qj}) | (p, q) \in \alpha\} \subset L$ as a fuzzy set that captures our observation on the neighbourhoods of $v_i$ and $v_j'$ in terms of binary compatibility function $f(x_{pi}, x'_{qj})$ (assuming no unary measurement). Combining the prior knowledge $F_{ij}^0(\alpha)$ and the observation $F_{ij}^b(\alpha)$ we should be able to obtain an a posteriori estimate of the label function $l_{ij}(\alpha)$ under the Basic Joint Interpretation $\alpha$ of the neighbourhoods $U$ and $U'$. Note that both $F_{ij}^0(\alpha)$ and $F_{ij}^b(\alpha)$ are fuzzy sets on the same universe of discourse $L$. All we have to do now is to find a formula that fuses the information from the two fuzzy sets. $F_{ij}^b(\alpha)$ and $F_{ij}^0(\alpha)$ represent the influence of the basic interpretation $\alpha$ of the neighbourhood to the label function $l_{ij}(\alpha)$. When the influence represented in the two sets contradicts (cancel) each other, we expect less support from the neighbourhood interpretation $\alpha$ to the label assignment $(v_i, v_j')$ and therefore a smaller value of $l_{ij}(\alpha)$. When the prior label assignment agrees with observational evidence, we expect a strong support to $l_{ij}$ from the joint neighbourhood interpretation $\alpha$. Now we can see that the information fusion function that we need to find in order to combine evidences in fuzzy sets $F_{ij}^b(\alpha)$ and $F_{ij}^0(\alpha)$ should measure the degree of match between the sets.

We propose an evidence combining formula in the form of cardinality and fuzzy set intersection

$$M(F \cap F')$$

where $F$ and $F'$ are the two fuzzy sets containing the evidences, $M()$ measures the cardinality of a fuzzy set and can be computed by summing up all the dimensions of the vector representation of the fuzzy set. One can see the function does measure the degree of match of the two fuzzy sets. Specifically it yields zero when $F$ and $F'$ are disjoint, and yields maximum value when $F$ and $F'$ maximally overlap.

Fuzzy set theory uses the minimum operator to define set intersection(c.f.[13]). Accord-
6.3. A RELAXATION LABELLING ALGORITHM BASED ON FUZZY SET THEORY AND ITS NON-ITERATIVE VARIANT

\[ Q_{ij}(\alpha) = M(F^b_{ij}(\alpha) \cap F^a_{ij}(\alpha)) = \sum_{(p,q) \in \alpha} \min[f(x_{pi}, x'_{qj}), l_{pq}] \quad (6.4) \]

where \( Q_{ij}(\alpha) \) is the value of \( M(F^b_{ij}(\alpha) \cap F^a_{ij}(\alpha)) \) viewed as a function of \( i \) and \( j \) and \( \alpha \). Note that \( Q_{ij}(\alpha) \) is different from the posterior label \( l_{ij}(\alpha) \) that we want to compute in the sense that \( Q_{ij}(\alpha) \) is not bounded in \([0, 1]\). Therefore some form of normalisation is needed. In conventional approaches as discussed in the previous section, partition functions are used for normalisation. The result is a bounded label function but with a lot of side effects. Here we propose a different normalisation that is free of those problems.

Note that we have assumed that \( \alpha \) is a Basic Joint Interpretation on the neighbourhoods \( U \) and \( U' \) of size \( n \) and \( n' \) respectively. Therefore set \( \alpha \) in \( F^b_{ij}(\alpha) = \{f(x_{pi}, x'_{qj})|(p,q) \in \alpha\} \) and \( F^a_{ij}(\alpha) = \{l_{pq}|(p,q) \in \alpha\} \) should describe a one-to-one mapping. Let the cardinality of the set of interpretations \( \alpha \) be \( M(\alpha) \), then \( M(\alpha) \) is equal to the minimum of the size of the two neighbourhoods. Assuming \( n \leq n' \), then \( M(F^a_{ij}(\alpha) \cap F^b_{ij}(\alpha)) \leq M(\alpha) = n \). When all labels in \( F^a_{ij}(\alpha) \) and \( F^b_{ij}(\alpha) \) are positively certain, \( M(F^a_{ij}(\alpha) \cap F^b_{ij}(\alpha)) \) yields maximum value \( n \). Therefore we propose to use \( n \) to normalise \( Q_{ij}(\alpha) \) in equation (6.4) and give the posterior label function:

\[ l_{ij}(\alpha) = \frac{Q_{ij}(\alpha)}{n} = \frac{\sum_{(p,q) \in \alpha} \min[f(x_{pi}, x'_{qj}), l_{pq}]}{n} \quad (6.5) \]

An interesting view of this normalisation is that it is in fact calculating the average of \( \min[f(x_{pi}, x'_{qj}), l_{pq}] \), for all \((p,q) \in \alpha\). As discussed in the last section, we see this averaging operation as finding a compromise among the evidences provided from different neighbours in the neighbourhood. As there is no further cue suggesting which assignment pair of \((p, q)\) should be favoured or penalised, it is a logical choice to select the compromise of these evidences. This is very different from the normalisation with partition function used in probabilistic relaxation and many other relaxation algorithms, since partition function style of normalisation always favours the best supported interpretation and penalises the less supported interpretation.

The number of possible Basic Joint Interpretations on \( U \) and \( U' \) is very large \( (\frac{n'!}{(n'-n)!}) \). Among them there may be more than one receiving high support from the prior information and the observations, i.e. \( l_{ij}(\alpha) \) maybe near 1 for a number of different \( \alpha \). As discussed in
Section 6.1, an arbitrary joint interpretation (which may contain one or many ambiguous label assignments) can be viewed as the disjunction of a number of non-ambiguous Basic Joint Interpretations. We improvise on this idea and take the fuzzy disjunction (max) of all label functions \( l_{ij}(\alpha) \) computed for different \( \alpha \)s as our final label estimate \( l_{ij} \). Now we have the label update formula:

\[
l_{ij}^{k+1} = \max_{\alpha} \{ l_{ij}^k(\alpha) \} = \max_{\alpha} \{ \sum_{(p,q) \in \alpha} \min [ f(x_{pi}, x'_{qj}), l_{pq}^k] \}
\]

(6.6)

where \( k \) is the number of iterations.

Since \( \alpha \) is a one-to-one mapping, a closer observation of (6.6) would reveal that the combinatorial maximisation on the dominator of the formula is equivalent to the Maximum Weight Bipartite Matching problem, which can be solved in polynomial time. In [14], an efficient algorithmic solution based on Fibonacci Heaps was given. The algorithm has complexity \( O((n + n') \times n \times n') \). For medium sized neighbourhoods, this is much more efficient than the combinatorially exhaustive evaluation of supports. However, for small neighbourhoods (less than 6 or 5), an exhaustive search would be simpler.

For very large neighbourhood, \( O((n + n') \times n \times n') \) complexity algorithm can still be quite slow. Therefore we propose a Mean Field Theory based algorithm for the maximisation problem in (6.6) as follows. Focusing on the neighbourhoods of \( v_i \) and \( v'_{j} \), let \( c_{pq} = \min [ f(x_{pi}, x'_{qj}), l_{pq}^k] \), let boolean variable \( X_{pq} = 1 \) represent the interpretation of \( v'_{q} \in U' \) with \( v_i \in U \) and \( X_{pq} = 0 \) represent there is no such interpretation. The goal is to find \( X_{pq} \) for all \( p \in [1,n], j \in [1,n'] \) that maximise objective function \( E_c \):

\[
\max \{E_c = \sum_{p=1}^{n} \sum_{q=1}^{n'} c_{pq} X_{pq} \} \iff \min \{-E_c = \sum_{p=1}^{n} \sum_{q=1}^{n'} -c_{pq} X_{pq} \}
\]

(6.7)

where \( X_{pq} \in \{0,1\} \). As bipartite matching assumes one-to-one mapping between all \( v_p \) and \( v'_q \), we represent this constraint to (6.7) in the form of integer programming:

\[
\forall q \in [1,n'], \sum_{p=1}^{n} X_{pq} = 1 \quad and \quad \forall p \in [1,n], \sum_{q=1}^{n'} X_{pq} = 1 \quad or \quad 0
\]

(6.8)
Let us construct the cost function for the minimisation objective in (6.7) as

\[
E = \sum_{p=1}^{n} \sum_{q=1}^{n'} -c_{pq}X_{pq} + \sum_{q=1}^{n'} \frac{k_1}{2} \left( \sum_{p=1}^{n} X_{pq} - 1 \right)^2
+ \sum_{p=1}^{n} \frac{k_2}{2} \left( \sum_{q=1}^{n'} X_{pq} \right)^2 \left( \sum_{q=1}^{n'} X_{pq} - 1 \right)^2 + \frac{k_3}{2} \sum_{p=1}^{n} \sum_{q=1}^{n'} X_{pq} \left( X_{pq} - 1 \right) \quad (6.9)
\]

where \(k_1, k_2\) and \(k_3\) are constants. The first term in (6.9) is the objective, the second term represents constraint \(\forall q \in [1, n']\), \(\sum_{p=1}^{n} X_{pq} = 1\), the third term represents constraint \(\forall p \in [1, n]\), \(\sum_{q=1}^{n'} X_{pq} = 0\) or \(1\), and the last term ensures \(X_{pq} \in \{0, 1\}\). Mean Field Approximation is used to solve the minimisation of the cost function in (6.9)

\[
X_{pq} = \frac{1}{2} (1 + \tanh(\frac{u_{pq}}{T})) \quad (6.10)
\]

where \(T\) is the annealing temperature and is reduced every iteration,

\[
u_{pq} = -\frac{\partial E}{X_{pq}} = c_{pq} - k_1 \left( \sum_{i=1}^{n} X_{iq} - 1 \right) - k_3 \left( \frac{1}{2} - X_{pq} \right)
- k_2 \left( \sum_{j=1}^{n'} X_{pj} \right) \left( \sum_{j=1}^{n'} X_{pj} - 1 \right) \left(2 \sum_{j=1}^{n'} X_{pj} - 1 \right) \quad (6.11)
\]

Note that the above annealing algorithm can be made more efficient than \(O((n + n') \times n \times n')\), since \(\sum_{q=1}^{n'} X_{pq}\) and \(\sum_{p=1}^{n} X_{pq}\) need to be evaluated only once per iteration. If we designed an annealing schedule that requires \(m \ll n, n'\) iterations to complete, the computational complexity the MFT based algorithm would be \(O(n \times n' \times m)\) which is more efficient than [14] for very large sizes of \(n\) and \(n'\). The overall complexity of the proposed relaxation process is \(O(N \times N' \times n \times n' \times m)\) for MFT based maximisation, and is \(O(N \times N' \times n \times n' \times (n + n'))\) for Fibonacci Heap based maximisation, where \(N\) and \(N'\) are the number of nodes in the two graph. Note that \(n, n'\) and \(m\) are generally much smaller than \(N\) and \(N'\) for problems with only local interaction. In such case the complexity of our algorithm approximates \(O(N \times N')\).

The sizes of the neighbourhoods \(n\) and \(n'\) represent the scale of interaction in graph \(G\) and \(G'\). When \(n\) and \(n'\) are large, evaluation of supports from neighbourhoods is exponentially complex in the formulation of the labelling problem using Bayesian theory [5] and
many existing ARG labelling algorithms, although in some cases the complexity can be reduced to polynomial by factorisation if completely connected ARG topology is assumed [1]. To alleviate the exponential complexity, “relaxation” is used to iteratively propagate the constraints, assuming that global constraints can be represented as a collection of independent local constraints. Our formulation of the ARG labelling problem in (6.6) has first illustrated a transformation of the exponentially complex task of evaluating the neighbourhood support into a Maximum Weight Bipartite Matching problem for which an optimal polynomial-time algorithm exists [15]. The implication is that Markovian Property does not have to exist, if we solve the labelling problem with formula (6.6) and compute the maximisation using the Fibonacci Heap method. In other words, one does not have to compromise large scale neighbourhood constraints by assuming they can be broken down into a collection of local constraints and adopt iterative refinement. The result is a non-iterative labelling algorithm:

\[
l_{ij} = \frac{\max_{a} \{\sum_{(p,q)\in a} f(x_{pi}, x'_{qj})\}}{n}
\]  

(6.12)

where the maximisation is evaluated by algorithms described in [15], [14] etc. One can easily see that (6.12) is in fact formula (6.6) applied for only one iteration, with prior label values assumed to be one for all label assignments. This non-iterative algorithm has complexity \(O(N \times N' \times n \times n' \times (n + n'))\).

Both iterative and non-iterative procedures proposed in (6.6) and (6.12) produce label values within \([0, 1]\) without using the partition function. Therefore our algorithms are free of all the side effects introduced by partition function, namely label amplification, label dilution and the need for null attractor. Multiple interpretations are naturally supported by our algorithm. The result of the labelling process is the same when we interchange the model and the scene. This is another property that does not exist in labelling algorithms that do not support multiple interpretations.

6.4 Experiments

Experiment 1 shows a 2D shape matching problem where a symmetric feature causes inherent multiple interpretations of objects. Experiment 2 is the road matching problem as seen in [1]. In this experiment, we show that our algorithm is able to perform complex structural
matching with large neighbourhoods efficiently.

Experiment 1 demonstrates extracting similar parallelograms from an indoor scene (figure 6.1). The similarity-invariant measurement used here is the same as in Experiment 2 (c.f. equation 6.13). Notice that some of the parallelograms are not complete. This is not an error since we label each side of the parallelogram separately, and it is possible that one or two of the side(s) of the parallelogram do not match well and get rejected. Since the model graph is quite small, it took less than 2 seconds for the algorithm to complete labelling all 244 edge segments in the scene. Multiple interpretation is happening in this problem. Parallelograms are inherently an ambiguous shape when we try to label their four sides. In addition, in figure 6.1 some of the line segments are shared by the nearby parallelograms, which can also cause label ambiguity.

Experiment 2 shows a road matching problem. Roads in the aerial images and maps are approximated by groups of line segments. The task is to identify a particular road section in the hand-drawn map from a given image. Here we choose each node of the ARG as a single line segment represented by the two end points. For each pair of line segments, a similarity (translation, rotation and scaling) invariant binary relation can be calculated using ratios of distances between the end points. Specifically, let $p_{1i}$ and $p_{2i}$ be the starting and ending points of the $i$th line segment, $p_{1j}$ and $p_{2j}$ be the starting and ending points of the $j$th line segment. Binary relation $x_{ij}$ is a 4 dimensional vector:

$$s_{ij} \cdot \frac{d(p_{1i}, p_{1j})}{d(p_{1i}, p_{2i})} \cdot \frac{d(p_{2i}, p_{1j})}{d(p_{2i}, p_{2i})} \cdot \frac{d(p_{2i}, p_{2j})}{d(p_{2i}, p_{2i})}$$

where $d(p_1, p_2)$ is the Euclidean distance between point $p_1$ and $p_2$. $s_{ij}$ is the sign function and equals 1 if $p_{1i}, p_{2i}$ and $p_{1j}$ are arranged in the clockwise order or collinear, and equals -1 if not. Unlike angle based measurements, feature vector in (6.13) is a normal form, and can describe relations between arbitrary pair of line segments even when their ends are joining together or overlapping. Notice that the feature vector in (6.13) yields different values if we interchange the starting point(s) and the ending point(s). Since we do not know in which direction the line segments are drawn in the image and the map, we have to accommodate both interpretations by assuming an extra “reversed match”. This can be easily done by adding a new node (which is the reversed line segment) for each line segment either in the map or in the image. Although the number of nodes in one of the ARG has doubled, our
6.5 Summary and Conclusion

Partition function of the form (6.1) has been widely used by many relaxation labelling algorithms. In section 2, we described four types of problems introduced by the partition function and argued that the lack of a genuine compromise operator is the fundamental reason...
why partition function was used. In section 3, we derived a new evidence combining formula (6.5) from fuzzy set theory to directly address the problem. This formulation enables us to transform the combinatorial evaluation of neighbourhood support into a Maximum Weight Bipartite Matching problem for which an optimal polynomial-time algorithm exists. Without the need for Markovian Property, a non-iterative (non-relaxational) algorithm(6.12) was also made possible by the formulation. We demonstrated the effectiveness of our algorithms with a shape recognition problem and a road matching problem, In particular the first experiment has shown our algorithm's capability of handling multiple interpretations.
Bibliography


Chapter 7

Generating 2D and 3D global hypotheses from 2D affine correspondences

In Chapter 2 and 4, we described how to extract 2D surface features from the image. Chapter 3 and 5 described methods for constructing distributed 2D affine invariant representations of the surface patterns. Each connected component of the ARG representation of an object represents a single surface. All measurements are assumed to be taken independently from individual surfaces, and there is no 3D knowledge employed in the matching. In this chapter, we introduce 3D relations further to constrain the labels found previously and recover the 3D pose from the 2D labels.

As we argued in the previous chapter, ambiguities are commonplace in matching of 2D surface patterns. In a knowledge-based vision system, such ambiguities should be left to be resolved when additional information is available, and should not be blindly “ridiculed” (penalised) by a labelling scheme that assumes a closed-world of a set of models. The introduction of 3D relational information can often help to resolve the ambiguity seen by a limited 2D world model. If the labels provided by the 2D reasoning module can neither express multiple possibilities nor represent degree of commitment to each label faithfully according to the evidence, then it will be impossible for us to introduce more evidence later when it becomes available. Such modules will not be able to collaborate successfully with
other knowledge sources.

From the specific knowledge source point of view, vision is often an under-determined (ill-posed) problem. Successful and fast recognition can only result from a parallel or cascaded collaboration of multiple knowledge sources, each accommodating and utilising others' inputs.

In this chapter we first illustrate how to extract and refine independent global affine hypotheses from the model-object correspondences given by our 2D matching scheme. Such correspondences manifest themselves through a set of label probabilities or fuzzy membership function values, which may include multiple interpretations and multiple instances of the same model, i.e. it is a probabilistic/fuzzy many-to-many mapping. It is often sufficient to generate an affine hypothesis from single, pairwise or triplet label assignments, and there can be many hypotheses even after pruning by thresholding the label values. However note that many of these hypotheses actually belong to the same joint interpretation and hence amount to the same global affine transform. Therefore our first task is to extract independent global hypotheses using the probabilistic (or fuzzy) label values and to refine the affine parameters of each joint interpretation independently.

In Section 2 we introduce the techniques of recovering 3D pose from a set of independent 2D affine hypotheses of the surfaces using a single calibrated camera.

### 7.1 Recovering global affine transform from label messages

The object centred matching algorithms introduced in the last chapter typically generate a matrix of posterior label probabilities (or fuzzy label values) of the form \( l_{ij} \in [0, 1] \), where \( i \in [1, n] \) is the index of objects in the scene and \( j \in [1, n'] \) is the index of the objects in the model. Now we describe how to use the correspondence information given in \( l_{ij} \) to deduce the global affine transform that will project all the objects in the model into the scene coordinate system and yield a maximum overlap between the matched regions.

In Chapter 5, we described methods of recovering affine transform from the correspondence of regions. An affine transform can be uniquely determined if we know the correspondence \( R \leftrightarrow R', p \leftrightarrow p' \) where \( R \) and \( p \) are a region and a point in the model domain and \( R' \) and \( p' \) are a region and a point in the scene domain. If the reference points \( p \) are
7.1. RECOVERING GLOBAL AFFINE TRANSFORM FROM LABEL MESSAGES

found using methods introduced in Section 5.2.7 from a set of regions only, then we can simply state that we can recover an affine transform from a single correspondence of regions ($R \leftrightarrow R'$). This method requires least correspondence information but it is more sensitive to error in the unary measurements of individual regions, such as those caused by occlusion. Alternatively we can recover the affine transform from pairs of regions using the correspondence ($R_1 \leftrightarrow R'_1, R_2 \leftrightarrow R'_2$). Moreover, if we use the correspondence of triplets of regions ($R_1 \leftrightarrow R'_1, R_2 \leftrightarrow R'_2, R_3 \leftrightarrow R'_3$), the affine transform can be recovered using the correspondence of their centroids only. Using pairs or triplets requires more correspondence information but the resulting transform parameters are less sensitive to noise and occlusion that affect individual regions. In general, area, centroid, second order moments and a reference point provide eight real measurements for each region. Using projection equations of these measurements (c.f. Section 5.2.1), we can construct eight equations for each correspondence provided by the matching algorithms. If, in total, $N$ correspondences are found, there can be as many as $8 \times N$ equations. Given that there are only six affine parameters, the system is over constrained.

7.1.1 Global transform from one to one correspondence

If we assume that there is only one possible projection from the model to the scene (i.e. there is only one object in the image), then the over constrained system of equations can be solved by the Least Squares Method. Suppose we know the correspondence of object model $i$ with object $s(i)$ in the scene. Let $x_i$ denote a measurement vector of $i$ and $x'_{s(i)}$ denote the corresponding measurement vector of $s(i)$. If $s(i)$ is related to $i$ by an affine transform $A$, then $x'_{s(i)}$ can be written as some function of $A$ and $x_i$. If we have $N$ correspondences, we have $N$ equations

$$x'_{s(i)} = f(A, x_i) \quad i \in [1, N]$$

(7.1)

The Least Square method attempts to find $A$ that minimises quantity

$$J_A = \sum_{i=1}^{N} (f(A, x_i) - x'_{s(i)})^2$$

(7.2)
7.1. RECOVERING GLOBAL AFFINE TRANSFORM FROM LABEL MESSAGES

Since we already know the certainty (or probability) $l_{ij}$ of each label assignment $i \leftrightarrow j$, we propose to use a Weighted Least Square criterion:

$$J_x = \sum_{i=1}^{n} \sum_{j=1}^{n'} (f(A, x_i) - x'_j)^2 l_{ij}$$  \hspace{1cm} (7.3)$$

where $n$ is the number of objects in the model, and $n'$ is the number of objects in the scene. Minimising $J_x$ will find an $A$ that maximally satisfies the set of correspondences given in $l_{ij}$ with respect to feature vector $x$. Note that in Chapter 5, we already discussed the Least Square method as a possible way of solving simultaneous non-linear equations. We argued that such method of affine recovery is slow and prone to be trapped in local minima of the cost function. The problem with the Least Square method is that it is a directed search scheme based on local information $\frac{\partial J_x}{\partial A}$. It is difficult to use such a method directly for a coarse level search and the generation of initial hypothesis (which is our goal in Chapter 5) without a good initial guess. Here the situation is different because we can easily generate a good initial guess of affine transform by using unary, pairwise or triplet correspondences of regions. Our Least Square search is reliable if we start at a near optimal point, and $J_x$ is locally smooth. In fact if we construct $J_x$ using area, 2nd order moments and centroids as described previously, the energy function will only contain 1st and 2nd order functionals of $A$ and therefore will be locally smooth. In practice we can start the search at a hypothesis which is the weighted sum of hypotheses generated by individual regions, region pairs or region triplets. For example, for a collection of corresponding regions pairs $(R_i \leftrightarrow R'_s(i), R_p \leftrightarrow R'_s(p))$, if the affine transform recovered from the pair is $A_{ip}$, then the initial hypothesis is

$$A = \frac{\sum_{i,p} l_{is(i)} l_{ps(p)} A_{ip}}{\sum_{i,p} l_{is(i)} l_{ps(p)}}$$  \hspace{1cm} (7.4)$$

7.1.2 Multiple global transforms from many to many correspondence

The Least Square method collapses when there are multiple instances of a model object in the scene or there are multiple interpretations of the same object. Consider the case when there are two instances of the same object. Unless the two instances are coplanar, there is no single global transformation that can simultaneously satisfy both sets of correspondences. The minimisation of $J_x$ introduced previously will yield a compromise between the two
possible transforms. Ideally we would like to obtain two separate transforms in such cases. We shall now introduce the methods that can handle this situation properly.

Remember that our goal is to extract a global transform that produces a maximal match between two sets of regions, one in the model domain, the other in the image domain. The first possible way is to use a hypothesis-and-verification method. By sequentially selecting the corresponding regions, region pairs or region triplets, we can generate one affine hypothesis at each time and verify it by back-projecting the model into the image domain using the deduced transform and measure the difference. We can find several hypotheses as the local minima of

$$J(A) = \sum_{p=(x,y) \in M} (m(p) - s(Ap))^2$$

where $A$ is an affine transform hypothesis that maps model point $p$ to scene point $Ap$, and $M$ is the set of all the points in the model. $m(x, y)$ and $s(x, y)$ are the image functions of model and scene respectively. Finding minima of $J(A)$ by evaluating all possible $A$ is expensive especially if we choose to use pairs or triplets of regions to generate hypotheses. For each hypothesis, evaluating $J(A)$ by calculating the projection and difference for each model point is also time-consuming. More importantly, since the hypotheses are generated based on only one or two or three regions, the error can also be quite large. For most of the time, hypotheses generated this way require a further improvement using e.g. Least Square methods.

Another way of finding multiple optimal affine transforms from a set of correspondences is to use hypothesis clustering [1] [2] [3] [4]. This involves generating all hypotheses from the correspondences and discover one or more significant clusters in the hypothesis space. This method is faster than hypothesis-and-verification. To distinguish strongly supported hypotheses from those weakly supported, we need to weight every hypothesis with the related label assignment probability $l_{ij}$ in a way similar to 7.4. The challenge of using this approach is to model the cluster of affine transform hypotheses accurately under a variety of conditions. Usually the error in the hypotheses is due to partial occlusion, region erosion and image quantisation. These phenomena often yield a consistent bias in affine parameters of all hypotheses, e.g. all scaling parameters will be smaller than the actual value when region erosion happens. In such cases the cluster centre, as a weighted average of all hy-
7.1. Recovering Global Affine Transform from Label Messages

Potheses, might not be a good estimator of the global transform. Again a Least Square pose optimisation might be required after obtaining the clusters.

7.1.3 Multiple global transforms from a set of Basic Joint Interpretations

We shall now describe a novel strategy for finding the set of optimal affine transforms under many-to-many mapping. This method employs the optimal contextual correspondence provided by the labelling algorithm introduced in Chapter 6. The algorithm is fast because it can directly obtain the global transformations that are associated with different joint interpretations without clustering, or exhaustively generating and testing of hypotheses.

As described in Chapter 6, we can use Maximum Weight Bipartite Matching to find optimal one to one correspondences for the neighbourhood of an arbitrary node i. Each one to one correspondence describes a Basic Joint Interpretation of the neighbourhood. Using our proposed ARG representation, two label assignments will have high compatibility (i.e. similar binary measurements in the model and scene domain) if and only if the two model nodes undergo the same affine transform (c.f. Chapter 3 Section 2, and Chapter 5). Therefore each Basic Joint Interpretation uniquely corresponds to one possible global affine transform from a set of nodes in the model to a set of nodes in the image. Extracting all possible global affine transforms under the many-to-many mappings described by $l_{ij}$ is equivalent to finding all possible Basic Joint Interpretations that have a high total sum of probabilities. According to our labelling formula 6.12, $l_{ij} = \max_\alpha \left( \sum_{i \in O, j \in A} f(x_{pi}, x'_{pj}) \right)$ will be large if and only if there is at least one Basic Joint Interpretation that involves label assignment $i \leftrightarrow j$. Therefore we can simply store all the optimal Basic Joint Interpretations found by the Maximum Weight Bipartite Matching for each $i$ and $j$, and use them later for recovering the global affine parameters.

Note that some of the label assignments will generate the same BJIs. Duplicated interpretations only need to be stored once. In the clustering based methods, this is achieved by partitioning the set of transforms into clusters. Here since we are dealing with discrete joint label assignments, we can test for identity of BJIs instead of measuring the distance between different affine transforms which is a somewhat difficult concept to define mathematically. Moreover, the elimination of duplicate BJIs can be done in logarithmic time with respect to the number of existing BJIs when we introduce a simple linear order to the finite countable
set of BJIs and store them in a sorted sequence. Normally, there are only a few plausible
global transforms, including the cases of multiple interpretations of objects and multiple
instances of models. The maintenance of the sorted sequence of BJIs is very efficient.

Given a Basic Joint Interpretation, we can derive an optimal affine transform that satisfies
all correspondences in it. Because the mapping is one to one, the Least Square method
introduced earlier can be used to improve the hypothesis.

Obtaining a set of plausible global 2D affine hypotheses allows us to reconstruct all sur­
face features that previously would not be detected (such as surface boundaries) by project­
ing the 2D surface model into the image plane.

7.2 Pose recovery from a 2D affine transform hypothesis

7.2.1 Scaled Orthographic Projection and affine transforms

So far our model of surface patterns is 2D, and the projection between the model and image
is described as a 2D affine transform

\[
\begin{align*}
x'_a &= ax + by + t_x \\
y'_a &= cx + dy + t_y
\end{align*}
\]

(7.6)

(7.7)

with six parameters \((a, b, c, d, t_x, t_y)\). The 2D affine transform model is in fact an approxima­
tion of the perspective projection for planar objects. After 3D rotation and translation, 3D
point \((x, y, z)\) is transformed to

\[
\begin{pmatrix}
X \\
Y \\
Z
\end{pmatrix} =
\begin{pmatrix}
r_{00} & r_{01} & r_{02} \\
r_{10} & r_{11} & r_{12} \\
r_{20} & r_{21} & r_{22}
\end{pmatrix}
\begin{pmatrix}
x \\
y \\
z
\end{pmatrix} +
\begin{pmatrix}
T_x \\
T_y \\
T_z
\end{pmatrix}
\]

(7.8)

Perspective projection is then applied to map the 3D object onto the 2D image plane, assum­
ing a camera centred coordinate system.

\[
\begin{align*}
x' &= \frac{f}{z_r}(x_r - X_0) + x_0 \\
y' &= \frac{f}{z_r}(y_r - Y_0) + y_0
\end{align*}
\]

(7.9)
where \( f \) is the focal length of the projection, which is assumed to remain constant. Suppose there is a set of \( N \) points \( \{(x_i, y_i, z_i) | i \in [1 \ldots N]\} \) being rotated, translated in 3D and then projected onto the image plane using the above formula. Assuming the depth coordinates \( z_i \) of each point are not very different from each other, i.e. \( z_i = z_0 \) is constant, then the perspective projection formula turns into a Scaled Orthographic Projection (SOP, also known as Weak Perspective Projection):

\[
\begin{align*}
x' &= s(x_r - X_0) + x_0 \\
y' &= s(y_r - Y_0) + y_0
\end{align*}
\]

where \( s = \frac{f}{f_0} \), and \((X_0, Y_0, Z_0)\) and \((x_0, y_0, f)\) are the origin of the object and the image plane in the camera centred coordinate system. Substituting 7.8 into 7.11 we have:

\[
\begin{align*}
x' &= s(r_{00}x + r_{01}y + r_{02}z + T_x - X_0) + x_0 \\
y' &= s(r_{10}x + r_{11}y + r_{12}z + T_y - Y_0) + y_0
\end{align*}
\]

If for all \( i, (x_i, y_i, z_i) \) lies on the same plane \( z = \alpha x + \beta y + \gamma \) the we have:

\[
\begin{align*}
x' &= s(r_{00}x + r_{01}y + r_{02}(\alpha x + \beta y + \gamma) + T_x - X_0) + x_0 \\
&= (sr_{00} + sr_{02}\alpha)x + (sr_{01} + sr_{02}\beta)y + s(r_{02}\gamma + T_x - X_0) + x_0 \\
&= ax + by + t_x \\
y' &= s(r_{10}x + r_{11}y + r_{12}(\alpha x + \beta y + \gamma) + T_y - Y_0) + y_0 \\
&= (sr_{10} + sr_{12}\alpha)x + (sr_{11} + sr_{12}\beta)y + s(r_{12}\gamma + T_y - Y_0) + y_0 \\
&= cx + dy + t_y
\end{align*}
\]

which is an affine transform.

7.2.2 Recovering the pose from the planar affine transform

In the last section we showed how to obtained a set of global affine transformations each corresponding to an independent hypothesis of an object in the scene. This section describe
how to resolve the 3D position and orientation of the objects from the affine parameters 
(a, b, c, d, tx, ty).

Several formulas of 3D pose recovery from 2D correspondences have been proposed by 
researchers [7] [9] [10] [11] [8] [5]. Here we also give a derivation. The difference between 
our solution and the previous methods is that our solution for depth and other parameters 
is given in terms of the six 2D affine parameters (a, b, c, d, tx, ty). Instead of using point by 
point correspondences, the depth and other 3D parameters are directly obtained from the 
estimated 2D affine transforms. The existence and uniqueness of the solution for depth is 
proved directly from the geometry of Scaled Orthographic Projection.

First let us ignore the translation parameters and see how to recover the parameters of a 
Scaled Orthographic Projection and 3D rotation. Figure 7.1 illustrates an SOP projection of 
four coplanar points (0, 0) (0, 1), (1, 1) and (1, 0) which are scaled by factor s and rotated in 
3D. Let the corresponding affine mapping be:

\[
\begin{align*}
    x' &= ax + by \\
    y' &= cx + dy
\end{align*}
\] (7.13)

Note that since 3D rotation and scaling does not change the position of origin, translation 
parameters tx and ty are zero here. Substituting the point coordinates into the above equa­
tion, we obtain the projection of the four points at (0, 0), (a, b), (c, d) and (a + c, b + d). Since 
the object is scaled by a factor of s, from Figure 7.1 we have the following relations

\[
\begin{align*}
    a^2 + c^2 + z_1^2 &= s^2 \\
    b^2 + d^2 + z_2^2 &= s^2 \\
    (a + b)^2 + (c + d)^2 + (z_1 + z_2)^2 &= 2s^2
\end{align*}
\] (7.14) (7.15) (7.16)

Subtracting 7.14 and 7.15 from 7.16, we have

\[
z_1 = -\frac{ab + cd}{z_2}
\] (7.17)

Substituting the above equation into 7.14 and eliminating z_2 using \(z_2^2 = s^2 - b^2 - d^2\) from
7.2. POSE RECOVERY FROM A 2D AFFINE TRANSFORM HYPOTHESIS

7.15, we have a biquadratic equation for \( s \) :

\[
s^2 - (a^2 + b^2 + c^2 + d^2) + (b^2 + d^2)(a^2 + c^2) - (ab + cd) = 0 \tag{7.18}
\]

Solving the equation and choosing the positive \( s \), we have

\[
s = \sqrt{\frac{1}{2}[a^2 + b^2 + c^2 + d^2 \pm \sqrt{(a^2 + b^2 + c^2 + d^2)^2 - 4(ad - bc)^2}]} \tag{7.19}
\]

Note that from 7.14 and 7.15, we have

\[
s^2 - (a^2 + c^2) = z_1^2 \geq 0 \tag{7.20}
\]
\[
s^2 - (b^2 + d^2) = z_2^2 \geq 0 \tag{7.20}
\]

Combining the inequalities and rearranging, we have

\[
s \geq \sqrt{\frac{1}{2}(a^2 + b^2 + c^2 + d^2)} \tag{7.21}
\]

Therefore we conclude that the scale parameter of SOP has one unique solution :

\[
s = \sqrt{\frac{1}{2}[a^2 + b^2 + c^2 + d^2 \pm \sqrt{(a^2 + b^2 + c^2 + d^2)^2 - 4(ad - bc)^2}]} \tag{7.22}
\]

From 7.14 and 7.15 we can find the relative depths \( z_1 \) and \( z_2 \) :

\[
(z_1, z_2) = \pm(\sqrt{s^2 - a^2 - c^2}, \sqrt{s^2 - b^2 - d^2}) \tag{7.23}
\]

Note that there are two possible relative depths for each point: one positive and one negative, except for the origin \((0, 0)\). This reflects the inherent ambiguity that the plane maybe flipped with respect to a plane that passes the origin and is parallel with the image plane. For an arbitrary point \((x, y)\) in the object centred plane defined by \((0, 0), (0, 1)\) and \((1, 0)\), its 3D coordinate (without translation) is

\[
\frac{1}{s} \begin{pmatrix} ax + by \\ cx + dy \\ xz_1 + yz_2 \end{pmatrix} = \frac{1}{s} \begin{pmatrix} ax + by \\ cx + dy \\ \pm(x\sqrt{s^2 - a^2 - c^2} + y\sqrt{s^2 - b^2 - d^2}) \end{pmatrix} \tag{7.24}
\]
From $s = \frac{f}{z_0}$ we have the coordinate of the object origin $(0, 0, 0)$ in the camera centre coordinate system as:

$$\begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} = \frac{1}{s} \begin{pmatrix} t_x - x_0 \\ t_y - y_0 \\ f \end{pmatrix} \quad (7.25)$$

Given a point $(x, y)$ in the model (object) centred coordinate system, the 3D position of the corresponding point in the camera centred coordinate system is:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \frac{1}{s} \begin{pmatrix} ax + by + t_x - x_0 \\ cx + dy + t_y - y_0 \\ f \pm (x\sqrt{s^2 - a^2 - c^2} + y\sqrt{s^2 - b^2 - d^2}) \end{pmatrix} \quad (7.26)$$
7.2. POSE RECOVERY FROM A 2D AFFINE TRANSFORM HYPOTHESIS

Figure 7.1: Scaled Orthographic projection of point (0,0) (0,1),(1,0) and (1,1)
7.3 Experiments

In this section we describe experiments that illustrate how the best affine hypothesis can be selected using the method introduced in section 7.1.3. First we segment the scenes into a collection of regions using the model based color segmentation algorithm described in the section 4.4.3. The model object used in this experiment is shown in Figure 7.2, specifically, the features used for matching and hypothesis generation are shown in Figure 7.3, which consist four regions of distinct shape.

Figure 7.4 to Figure 7.14 show the segmentation of the scene into regions before matching is performed.

After matching, a number of affine hypothesis are generated for each scene, these are illustrated in Figures 7.16, 7.18 7.20, 7.22, 7.24 and 7.26. As described in sections 7.1.2 and 7.1.3, traditionally, these hypothesis are then processed by clustering algorithms and/or hypothesis-verification process in order to eliminate duplicate hypothesis and discover the optimal hypothesis.

In section 7.1.3, we introduced a different approach that leverage the Basic Joint Interpretations that is made available by our Fuzzy ARG Matching Algorithm (Section 6.3). This means one can readily identify co-related and independent hypothesis in a many to many mapping scene without resorting to clustering and hypothesis verification. One can also compute more robust measurements that are previously not available in absence of the Joint independent, such that the transformation parameters can be reconstructed more accurately. For example, when triplets of points are used to compute affine transforms, it is known that the result is more reliable when the area of enclosed by the triplets of points are large (see Section 3.3.3). In our experiments, without a Joint Interpretation, one can only compute the transformation hypothesis individual regions, which are prone to error. Where as after the matching, when the joint labeling are available to groups of regions, we can compute features involving several regions and covers much larger area. Thus the transform parameters obtain tends to be more stable and accurate.

Figures 7.17, 7.19 7.21, 7.23, 7.25 and 7.27 show that the best transform hypothesis obtained using the best supported joint labelling. One can see that the errors in these cases are smaller compared to pictures on the left column. Note that we can also use least square based methods to further optimize the transformation parameters.
7.3. EXPERIMENTS

Figure 7.2: the Model Object

Figure 7.3: Regions in the Model Object that are used to for matching and generation of pose hypothesis. These regions are characterized by their color which is modeled using a simple Gaussian chromaticity distribution.

Figure 7.4: Scene 1.1

Figure 7.5: Regions in Scene 1.1 that are used to for matching and generation of pose hypothesis. These regions are extracted using the color segmentation algorithms described in Chapter 4.
Figure 7.6: Scene 1.2

Figure 7.7: Regions in Scene 1.2 that are used to for matching and generation of pose hypothesis.

Figure 7.8: Scene 2.1

Figure 7.9: Regions in Scene 2.1 that are used to for matching and generation of pose hypothesis.

Figure 7.10: Scene 2.2

Figure 7.11: Regions in Scene 2.2 that are used to for matching and generation of pose hypothesis.
7.3. EXPERIMENTS

Figure 7.12: Scene 3.1

Figure 7.13: Regions in Scene 3.1 that are used to for matching and generation of pose hypothesis.

Figure 7.14: Scene 3.2

Figure 7.15: Regions in Scene 3.2 that are used to for matching and generation of pose hypothesis.

Figure 7.16: Affine hypothesis for Scene 1.1: generated using the matching result of the algorithms described in Chapter 6

Figure 7.17: The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3
7.3. EXPERIMENTS

Figure 7.18: Affine hypothesis for Scene 1.2: generated using the matching result of the algorithms described in Chapter 6

Figure 7.19: The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3

Figure 7.20: Affine hypothesis for Scene 2.1: generated using the matching result of the algorithms described in Chapter 6

Figure 7.21: The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3

Figure 7.22: Affine hypothesis for Scene 2.2: generated using the matching result of the algorithms described in Chapter 6

Figure 7.23: The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3
7.3. EXPERIMENTS

Figure 7.24: Affine hypothesis for Scene 3.1: generated using the matching result of the algorithms described in Chapter 6

Figure 7.25: The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3

Figure 7.26: Affine hypothesis for Scene 3.2: generated using the matching result of the algorithms described in Chapter 6

Figure 7.27: The chosen affine hypothesis using the Best Supported Basic Joint Interpretation Criteria described section 7.1.3
7.4 Conclusion

At the end of the ARG matching process, we have obtained many-to-many correspondences between the model features and scene features, the next step is to retrieve the affine hypothesis from the mapping. The over-determined system of equations involved in "transformation from correspondences" problem can be solved by the Least Square Method. Unfortunately the method only works if the correspondence is one-to-one and good initial guess is available. Traditional methods such as Hypothesis Clustering, Hypothesis-and-Verification are able to deal with many-to-many mappings but also appear to be either reliant on knowledge about the underlying parameter clusters or sensitive to noise. In the first part of the Chapter 7, we showed that the fuzzy set theory based on which we developed our matching algorithm also provides a theoretical basis on which we can extract multiple 1-to-1 mappings from the label probabilities as multiple global interpretations (i.e. multiple affine hypothesis.) Our label message extraction method is closely tied with our object centered labelling algorithm such that it is able to reuse the intermediate computation results of the labelling algorithm. This makes our method very fast. The resulting affine transforms from the 1-to-1 correspondences can be optimized using standard least-square based methods.

To date, several formulas of 3D pose recovery from 2D point correspondences have been proposed. In the second part of the chapter, we also give a derivation. The difference between our solution and the previous ones is that our solution for depth and other parameters is given solely in terms of six 2D affine parameters estimated from the planar feature matching results. Instead of requiring point correspondences, our method only requires six affine parameters which can be estimated from correspondences of varieties of features such as regions, texture, curves.
Bibliography


156
Chapter 8

Summary and future works

8.1 Summary

In this work we mainly addressed three questions arising from the problem of recognising pattern-mapped polyhedrae objects, namely:

1. What type of representation do we use to represent our scenes and models in order to facilitate efficient and robust recognition.

2. How do we generate such representations for given scenes and models under projection invariant constraints.

3. How recognition (matching) can be achieved using such representations.

Our answer to the first question is to use Attributed Relational Graph representation of models and scenes. The reasons for our choice are that

1. ARG representation is a local representation, in other words every measurement used to constitute the representation can be extracted locally. This is advantageous over global representations because local representations as a whole are less susceptible to damage by occlusion and noise. Errors in the local measurements are kept at local level. The problem of Partial Inexact Matching of ARGs has so been formulated which further enhances the ability of the representation to cope with local measurement error.

2. Numerous algorithms based on Graph Theory, Relaxation and Stochastic Mechanics
have been developed to provide Efficient Partial Inexact Matching of Attributed Relational Graphs. This makes the recognition using the ARG representation feasible.

To construct the ARG representation, one needs first to break the scene/model into a set of local features (primitives) which will act as nodes of the graph. This is the stage of segmentation where the data volume is greatly reduced. In Chapter 2, we described edge based local features and techniques to extract them. We began with investigating the criterion functions for optimal feature detection, and proposed two new criteria that can help us in choosing saliency measurements and designing filtering processes. Unlike previous formulations, our criteria do not attempt to model the underlying filtering process. Instead, we focus our attention on the characteristics of the resulting signal and establish our objective functions based on these characteristics. The benefit of our approach is that it can be used to compare and optimise a much wider class of processing algorithms including adaptive and/or non-linear filters. We have also shown, in Chapter 2, the relations between our feature detection theory and the existing feature detection theories and filter designs. A simple novel adaptive filter was proposed and shown to give superior performance to existing filters.

Region is another type of primitive that we would like to use for constructing our ARG representation. Chapter 4 is dedicated to the problem of color/intensity based region segmentation. We centered our discussion, comparison and derivation of segmentation algorithms around the five spatial and feature domain constraints mainly proposed by Haralick. We illustrated that many problems found in the existing segmentation algorithms are due to some of the important constraints being overlooked or to the incorrect modelling of these constraints in the segmentation process. A new segmentation algorithm has been developed and shown to have overcome these problems and to provide better segmentation of color images. We also showed the relation of our segmentation algorithm with the Weak Membrane Model, and anisotropic filtering. Finally, a novel noise suppression filter based on our segmentation criteria was introduced.

Our main contribution in answering the second question lies in demonstrating a systematic way for deriving invariant measures that capture the object shape and global transformation constraints using unary and binary features only. Without having to invoke high order relations, the derived binary unary measures enabled us to employ efficient determin-
istic relaxation labelling methods to match our ARG image representation under projections. In Chapter 3, we introduced a framework for deriving the projective invariant measures using point features. Chapter 5 shows how to obtain affine invariant binary and unary measures using a mixture of point and moment features.

In Chapter 6, concerned with the problem of ARG matching, we first discussed the issue of a number of drawbacks in the existing deterministic ARG matching algorithms. We also argued that it is important for an object recognition system to be able to handle multiple interpretations (which has been a largely overlooked problem). We began our search for a labelling algorithm that is free of the problems of conventional algorithms by adopting a fuzzy set theoretic model of label interaction. We derived a label update formula that can be computed efficiently and its non-iterative variant. Our new labelling algorithm can handle multiple interpretations and express degree of commit in the final labels. In Chapter 7, we also showed how to extract joint interpretations from the map individual interpretations given by our labelling algorithm.

8.2 Future work

8.2.1 Feature detection using the criteria functions proposed in our Computational Theory of Feature Detection

At the beginning of Chapter 2, we introduced a new Computation Theory of Feature Detection. We derived a new filter (in section 2.5.2) using our theory. The objective function used in our filter in 2.5.2 is a simplified version of our original criteria. Apparently there can be many different ways of incorporating our criteria in an objective function, and many different ways of optimising such functions. It would certainly be interesting to see how one can derive more sophisticated adaptive filters using these methods.

The third criterion from Canny (suppression of multiple response) does not find a correspondence in our feature detection theory. Note that our theory is based on the idea of filtering the measurements for better post-processing by thresholding and non-maxima suppression. We can see that Canny’s third criterion is related to the non-maxima suppression process in some way. However it is not yet clear that how to describe the notion of maximising distances between peaks in terms of local differentials. It would be interesting to see
how one can further develop the theory to accommodate this criterion.

In 2.5.2, we have demonstrated the utility of our feature detection theory with the example of a novel adaptive corner detection filter. The similarity of edge detection and curvature feature detection detection suggests that the two processes are subject to the same principle and measure of quality. Therefore it would be very interesting to see how the theory could be extended into two dimensions and to develop novel adaptive filters for edge detection using the theory.

8.2.2 Modelling Haralick’s constraints in relaxation based segmentation algorithms

Our experience in developing color/grey-level image segmentation algorithms has shown us the importance of the five constraints proposed in Chapter 4 (mostly advocated by Haralick). In the later part of Chapter 4, we derived a color image segmentation algorithm using the five constraints. Our proposed method is obviously not the only way of integrating these constraints into a relaxation process. In fact one can find many different ways of expressing the constraints and quite possibly using an elegant formulation. Non-relaxation based methods can also improve their quality by taking into account these constraints, although we would like the emphasise that a soft decision making system might probably be better in integrating multiple constraints or combining multiple form of evidence.

8.2.3 Using multiple interpretations

The greatest strength of the new ARG matching algorithm proposed in Chapter 6 is the ability of handle multiple interpretations. We have argued that multiple interpretation is often needed when dealing with noisy and ambiguous images. It allows us to delay the decision making and we will not have to prune out the potential solutions until all the evidence is available. The degree of commitment made by the algorithm can be very useful in combining the results of multiple knowledge sources in a soft reasoning framework. Due to the limited time, we have not been able to show in this thesis an example of combining multiple knowledge sources and to demonstrate the strength of multiple interpretation. One immediate possibility is to cascade the 2D surface pattern matching with a 3D surface graph matching algorithm.