DOCTORAL THESIS

Interest Curves:

Concept, Evaluation, Implementation and Applications

BO LI

Digital Media Lab

Department of Applied Physics and Electronics

UMEÅ UNIVERSITY, SWEDEN

Umeå, 2015

Bo Li
Department of Applied Physics and Electronics
Umeå University
SE-901 87 Umeå, Sweden
Author e-mail: bo.li@umu.se

Thesis submitted for the degree of Doctor of Philosophy (Ph.D.)
in Applied Physics and Electronics with specialization in Signal Processing.

Akademisk avhandling för avläggande av teknologie doktorsexamen (Tekn. Dr.)
i tillämpad elektronik med inriktning mot signalbehandling.

ISSN: 1652-6295:21
ISBN: 978-91-7601-353-3
E-version available at http://umu.diva-portal.org
Printed by Print & Media, Umeå University, Umeå, Sweden, October, 2015.

Interest Curves:
Concept, Evaluation, Implementation, and Applications

Bo Li

Department of Applied Physics and Electronics, Umeå University, Sweden

ABSTRACT

Image features play important roles in a wide range of computer vision applications, such as image registration, 3D reconstruction, object detection and video understanding. These image features include edges, contours, corners, regions, lines, curves, interest points, etc. However, the research is fragmented in these areas, especially when it comes to line and curve detection. In this thesis, we aim to discover, integrate, evaluate and summarize past research as well as our contributions in the area of image features. This thesis provides a comprehensive framework of *concept*, *evaluation*, *implementation*, and *applications* for image features.

Firstly, this thesis proposes a novel concept of interest curves. Interest curves is a concept derived and extended from interest points. Interest curves are significant lines and arcs in an image that are repeatable under various image transformations. Interest curves bring clear guidelines and structures for future curve and line detection algorithms and related applications.

Secondly, this thesis presents an evaluation framework for detecting and describing interest curves. The evaluation framework provides a new paradigm for comparing the performance of state-of-the-art line and curve detectors under image perturbations and transformations.

Thirdly, this thesis proposes an interest curve detector (Distinctive Curves, *DICU*), which unifies the detection of edges, corners, lines and curves. DICU represents our state-of-the-art contribution in the areas concerning the detection of edges, corners, curves and lines. Our research efforts cover the most important attributes required by these features with respect to robustness and efficiency.

Interest curves preserve richer geometric information than interest points. This advantage gives new ways of solving computer vision problems. We propose a simple description method for curve matching applications. We have found that our proposed interest curve descriptor outperforms all state-of-the-art interest point descriptors (SIFT, SURF, BRISK, ORB, FREAK). Furthermore, in our research we design a novel object detection algorithm that only utilizes DICU geometries without using local feature appearance. We organize image objects as curve chains and to detect an object, we search this curve chain in the target image using dynamic programming. The curve chain matching is scale and rotation-invariant as well as robust to image deformations. These properties have given us the possibility of resolving the rotation-variance problem in object detection applications. In our face detection experiments, the curve chain matching method proves to be scale and rotation-invariant and very computational efficient.

Keywords: scale-invariance, edge, corner, curve, line, matching, object detection.

Sammanfattning

Bilddetaljer har en viktig roll i ett stort antal applikationer för datorseende, t.ex., bildregistrering, 3D-rekonstruktion, objektdetektering och videoförståelse. Dessa bilddetaljer inkluderar kanter, konturer, hörn, regioner, linjer, kurvor, intressepunkter, etc. Forskningen inom dessa områden är splittrad, särskilt för detektering av linjer och kurvor. I denna avhandling, strävar vi efter att hitta, integrera, utvärdera och sammanfatta tidigare forskning tillsammans med vår egen forskning inom området för bildegenskaper. Denna avhandling presenterar ett ramverk för *begrepp, utvärdering, utförande* och *applikationer* för bilddetaljer.

För det första föreslår denna avhandling ett nytt koncept för intressekurvor. Intressekurvor är ett begrepp som härrör från intressepunkter och det är viktiga linjer och bågar i bilden som är repeterbara oberoende av olika bildtransformationer. Intressekurvor ger en tydlig vägledning och struktur för framtida algoritmer och relaterade tillämpningar för kurv- och linjedetektering.

För det andra, presenterar denna avhandling en utvärderingsram för detektorer och beskrivningar av intressekurvor. Utvärderingsramverket utgör en ny paradigm för att jämföra resultatet för de bästa möjliga teknikerna för linje- och kurvdetektorer vid bildstörningar och bildtransformationer.

För det tredje presenterar denna avhandling en detektor för intressekurvor (Distinctive curves, DICU), som förenar detektering av kanter, hörn, linjer och kurvor. DICU representerar vårt främsta bidrag inom området detektering av kanter, hörn, kurvor och linjer. Våra forskningsinsatser täcker de viktigaste attribut som krävs av dessa funktioner med avseende på robusthet och effektivitet.

Intressekurvor innehåller en rikare geometrisk information än intressepunkter. Denna fördel öppnar för nya sätt att lösa problem för datorseende. Vi föreslår en enkel beskrivningsmetod för kurvmatchningsapplikationer och den föreslagna deskriptorn för intressekurvor överträffar de bästa tillgängliga deskriptorerna för intressepunkter (SIFT, SURF, BRISK, ORB, och FREAK). Dessutom utformar vi en ny objektdetekteringsalgoritm som bara använder geometri för DICU utan att använda det lokala utseendet. Vi organiserar bildobjekt som kurvkedjor och för att upptäcka ett objekt behöver vi endast söka efter denna kurvkedja i målbilden med hjälp av dynamisk programmering. Kurvkedjematchningen är oberoende av skala och rotationer samt robust vid bilddeformationer. Dessa egenskaper ger möjlighet att lösa problemet med rotationsberoende inom objektdetektering. Vårt ansiktsigenkänningsexperiment visar att kurvkedjematchning är oberoende av skala och rotationer och att den är mycket beräkningseffektiv.

Preface

The contributions presented in this thesis have previously been published, accepted or submitted to conferences and journals. A list of these publications is given below.

I. **Bo Li**, Shafiq Ur Réhman, Ulrik Söderström, "Independent Thresholds on Multi-scale Gradient Images", *in Proceedings of the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing (ICISIP 2013*), 26-27 September 2013, Kitakyushu, Japan. doi: 10.12792/icisip2013.027

II. **Bo Li**, Aleksandar Jevtic, Ulrik Söderström, Shafiq Ur Réhman, Haibo Li, "Fast Edge Detection by Center of Mass", *in Proceedings of the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing (ICISIP 2013*), 26-27 September 2013, Kitakyushu, Japan. doi: 10.12792/icisip2013.024

III. **Bo Li**, Ulrik Söderström, Shafiq Ur Réhman, Haibo Li, "Restricted Hysteresis Reduce Redundancy in Edge Detection", *in Journal of Signal and Information Processing*, August 2013. doi:10.4236/jsip.2013.43B028

IV. **Bo Li,** Haibo Li, Ulrik Söderström , "Scale-invariant Corner Keypoints", *IEEE International Conference on Image Processing*, October 2014.

V. **Bo Li**, Haibo Li, Ulrik Söderström, "Fast Edge Filter and Multi-scale Edge Detection", *Manuscript*.

VI. **Bo Li**, Haibo Li, Ulrik Söderström, "Distinctive Curve Features", *Submitted to Electronics Letters*.

VII. **Bo Li**, Haibo Li, Ulrik Söderström, "Distinctive Curves: Unified Scale-Invariant Detection of Edges, Corners, Lines and Curves", *Submitted to IEEE Transactions on Image Processing*.

VIII. **Bo Li**, Alaa Halawani, Haibo Li, Ulrik Söderström, "Scale & Rotation-Invariant Matching with Curve Chain", *Submitted to Computer Vision Journal*.

List of Publications

A list of all contributions published by the author is given below.

1. **Bo Li**, Shafiq Ur Réhman, Li, H. "i-Function of Electronic Cigarette: Building Social Network by Electronic Cigarette", *IEEE CPSCom 2011*, October 19-22, Dalian, China. doi: 10.1109/iThings/CPSCom.2011.124

2. **Bo Li**, “Integrated Emotion Module: Face Detection, Face Pose Estimation, Emotion Recognition”, Technical Report, INTRO Project (INTEractive RObotics Research Network).

3. Aleksandar Jevtic, **Bo Li**, “Ant Algorithms for Adaptive Edge Detection”, In: Search Algorithms Edited by: Taufik Abrão. InTech, ISBN:980-953-307-672-5, 2012.

4. **Bo Li**, Aleksandar Jevtic, Ulrik Söderström, Shafiq Ur Réhman, Haibo Li, “Fast Edge Detection by Center of Mass”, *in Proceedings of the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing (ICISIP 2013*), 26-27September 2013, Kitakyushu, Japan. doi: 10.12792/icisip2013.024

5. **Bo Li**, Ulrik Söderström, Shafiq Ur Réhman, Haibo Li, “Restricted Hysteresis Reduce Redundancy in Edge Detection”, *in Journal of Signal and Information Processing*, August 2013. doi:10.4236/jsip.2013.43B028

6. **Bo Li**, Shafiq Ur Réhman, Ulrik Söderström, “Independent Thresholds on Multi-scale Gradient Images”, *in Proceedings of the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing (ICISIP 2013*), 26-27 September 2013, Kitakyushu, Japan. doi: 10.12792/icisip2013.027

7. **Bo Li**, Licentiate Thesis: “*Pushing Edge Detection to the Limit — Towards Building Semantic Features for Human Emotion Recognition*”. Umeå University, December 2013.

8. **Bo Li,** Haibo Li, Ulrik Söderström, "Scale-invariant Corner Keypoints", *IEEE International Conference on Image Processing*, October 2014.

9. **Bo Li**, Haibo Li, Ulrik Söderström, "Fast Edge Filter and Multi-Scale Edge Detection", *Manuscript*.

10. **Bo Li**, Alaa Halawani, Haibo Li, Ulrik Söderström, "Scale & Rotation-Invariant Matching with Curve Chain", *Submitted to Computer Vision Journal*.

11. **Bo Li**, Haibo Li, Ulrik Söderström, "Distinctive Curve Features", *Submitted to Electronics Letters*.

12. **Bo Li**, Haibo Li, Ulrik Söderström, "Distinctive Curves: Unified Scale-Invariant Detection of Edges, Corners, Lines and Curves", *Submitted to IEEE Transactions on Image Processing*.

Acknowledgement

I feel grateful for the people who have helped me during my PhD studies. Without your support and patience, none of my work would be possible.

First of all, I wish to thank my supervisor, Professor Haibo Li, for giving me this research opportunity. Thank you for your encouragement, guidance, and patience. I am thankful for the free and passionate environment you build in our research group. Discussion with you is always a mysterious power which leads to a success of finding the correct path.

I am grateful to my supervisor, Dr. Ulrik Söderström, for all his support. You are always with me on the project journey. Thanks you for your endless support and company during the time that I grow as a researcher. I would like to thank my second supervisor, Professor Yael Edan, for her enthusiastic support during my research at Ben-Gurion University of the Negev. I would also like to thank Dr. Alaa Halawani, Dr. Shafiq Ur Réhman and Dr. Aleksandar Jevtic for great discussions, collaborations and contributions.

I am grateful to all the people in the INTRO project, for their friendly, energetic, and collaborative friendship. I wish to take this opportunity to extend my gratitude to everyone at the Department of Applied Physics and Electronics for their cooperation, understanding, and support which provides a pleasant working environment. I express my warm thanks to Annemaj Nilsson, Mona-Lisa Gunnarsson and Marie Fransson, who have always helped me.

Finally but most importantly, I owe sincere and earnest thankfulness to my parents and my wife. Without your great and selfless love, none of this would be possible.

Bo Li

October 2015

Contents

1 Introduction

1.1 Motivation

Image features play important roles in a wide range of computer vision applications, such as image registration, 3D reconstruction, object detection and video understanding. These image features include edges, contours, corners, regions, lines, curves, interest points, etc. However, previous research is fragmented in these areas, especially when it comes to line and curve detection. In fact, many recently developed computer vision applications still use very basic methods for the detection of edges, corners, lines or curves. Therefore, there is a real value of providing a clear, practical framework as a guide to the usage of these features.

In this thesis, we aim to discover, integrate, evaluate and summarize past research as well as our contributions regarding the detection of edges, corners, curves and lines. Therefore, this thesis provides a comprehensive framework including *concept*, *evaluation*, *implementation* and *applications* for features.

1.1.1 Concept

To fully understand, evaluate and summarize previous research and our contributions, we present a novel concept: *interest curves*. Most previous research in the area of line curve detection fall into this concept. Our proposal for the concept of interest curves brings the following research questions:

1. *What common goals are shared by line and curve detection methods?*
2. *How can we define the concept of interest curves?*
3. *What are the differences and similarities between interest curves and other types of features such as interest points and interest regions?*
4. *How do we design a standard interest curve data structure for all curve and line detection methods?*

1.1.2 Evaluation

Unlike interest points, which is a well-known concept, the concept of interest curves has not yet been proposed in previous research. Therefore, there is no comprehensive standard evaluation method and dataset available for interest curve detectors. Therefore, this thesis will answer the following questions:

1. *What parameters should we take into account in the evaluation of interest curve detectors, considering the concept and properties of interest curves?*
2. *How do we design a practical evaluation framework for the detection of interest curves?*

Interest curves can be used for feature correspondence tasks and related applications such as image stitching and 3D reconstruction. To solve the feature correspondence problem, a common approach is to describe local features as local descriptors and then measure the matching distance between the descriptors. Hence, finding a standard evaluation method for curve description and matching performance becomes an urgent issue. This issue forms the next question:

3. *How do we design a practical evaluation framework for interest curve descriptors?*

There are some differences and similarities between interest curves and interest points. Therefore, the next question is:

4. *How do we evaluate the detectors and descriptors for interest curves and interest points under the same evaluation framework?*

1.1.3 Implementation

The concept of interest curves and a new evaluation method brings a challenge to the previous algorithms in the areas of curves and lines detection. It is valuable to develop an interest curve detection algorithm and push the state-of-the-art to a new level. In comparison with interest points, the research in “interest curves” is not mature enough. Therefore, current curve and line detectors and descriptors are not able to present superior advantages. The first aim of our implementation is to:

- *Develop the state-of-the-art interest curve detection and description method.*

The performance of our method will be measured with the evaluation standard of interest curves.

In addition to stability performance, we are also interested in discovering more relationships between edges, corners, line and curves. The second aim of our implementation is to:

- *Develop an interest curve detector that can detect edges, corners and lines multi-functionally and efficiently*.

1.1.4 Applications

We use our state-of-the-art interest curve detection and description method in different applications to discover the unique advantages of our proposed interest curves. In addition, we have gone through previous research and have found the applications which can benefit from interest curves. Furthermore, we have developed new methods for object detection by utilizing the richer geometric information given by interest curves.

1.2 Thesis outline

The content of the thesis is organized in four chapters: concept, evaluation, implementation, and applications.

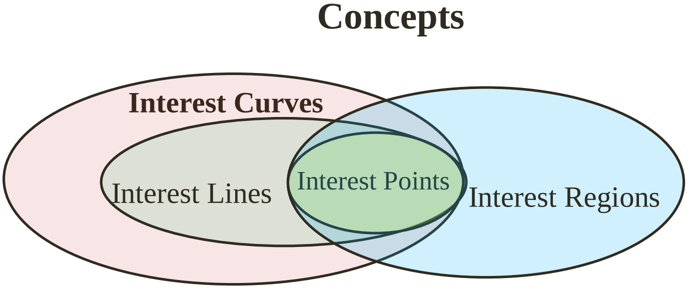


Fig. 1.1 Different concepts used in feature detection methods.

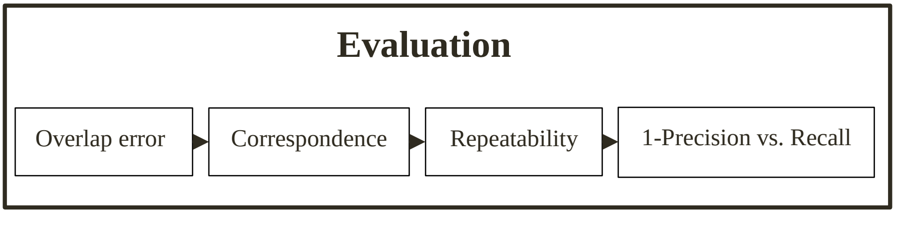


Fig. 1.2 Evaluation parameters.

Chapter 2 analyzes different concepts used in feature detection methods (**Fig. 1.1**) and presents our proposed concept of interest curves. This chapter also presents some common properties shared by various feature detectors. According to the proposed concept and findings, we propose a data structure and a guideline for algorithm design that can be used for interest curve detectors.

According to the attributes of interest curves, Chapter 3 proposes standard evaluation parameters (**Fig. 1.2**) and methodologies for both curve detectors and descriptors.

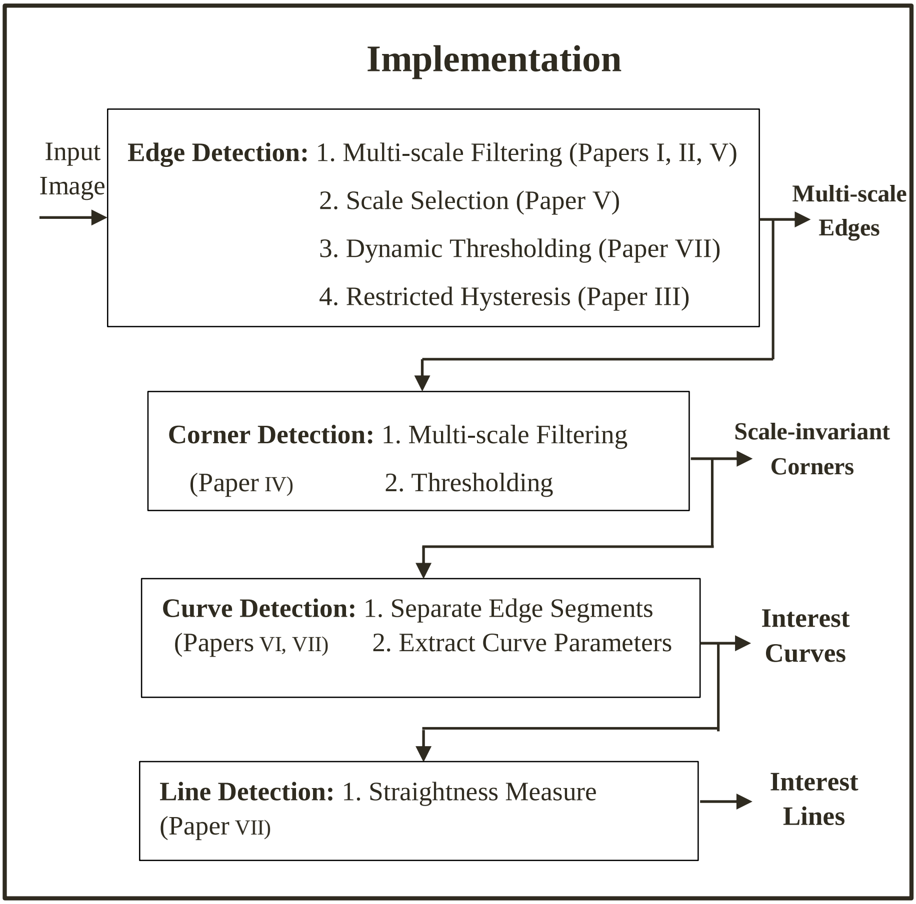


Fig. 1.3 Proposed multi-functional interest curve detector (Paper VII).

Chapter 4 presents our implementation of interest curve detection and description. It includes our contributions in edge detection, corner detection, curve and line detection and description. We present a unique multi-functional, efficient and robust curve detection method (**Fig. 1.3**).

Chapter 5 presents the application areas (**Fig. 1.4**) of interest curves and our novel curve-based object detection framework. Our curve-based object detection framework presents unique properties in the aspects of efficiency and rotation-invariance.

The concluding remarks are highlighted in Chapter 6. Finally, the research papers that provide the basis for this thesis are presented.

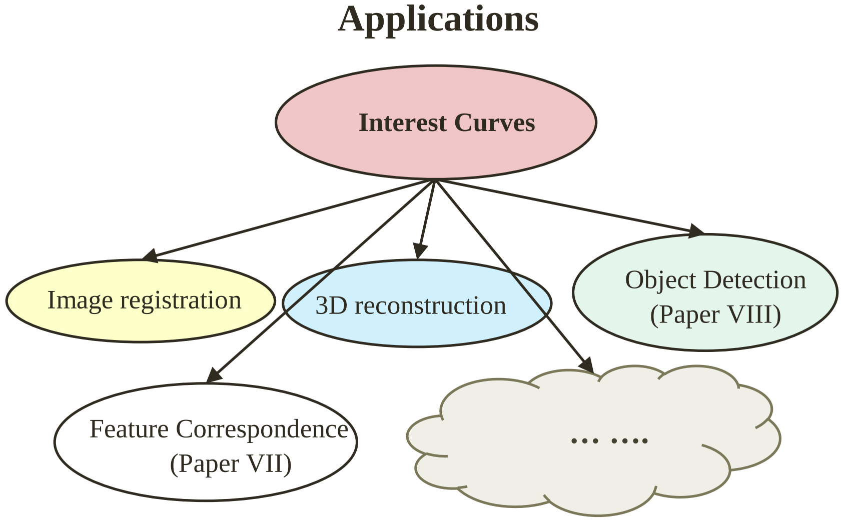


Fig. 1.4 Application areas of interest curves

2 Concept

In this Chapter, we analyze different concepts used in feature detection methods and present the concept of interest curves.

2.1 Background

Image features include edges [1], contours [2], corners [3, 4], lines [5, 6], curves [7], interest points [8, 9], region [10, 11], etc. These different types of features are categorized by geometrical concepts. Research in recent years has started to develop *interest point* features [9-12], which are repeatable in subsequent processing. An interest point is a point in the image which in general has the following characteristics:

- *It has a clear, preferably mathematically well-founded, definition.*
- *It has a well-defined position in the image space.*
- *It has (or is surrounded by) a distinctive local image structure.*
- *It is repeatable under various local and global perturbations in the image domain, such as illumination changes.*
- *Optionally but commonly, the notion of an interest point should include an attribute of scale, which makes it repeatable under image transformations including scale and view point changes.*

The interest point features used in early computer vision are actually corner features. The earlier corner detection methods [3, 4] aim to obtain robust, stable and well-defined points for object tracking and object recognition in a relatively simple environment. In the last decade, researchers have been more interested in detecting scale-invariant interest points [8, 9]. In practice, these interest point detectors are sensitive both to corners and image regions. In general, there is no rigorous boundary between interest points and interest regions. The reason is that an interest region can be represented as an interest point, because interest points includes the attribute of scale, which can reflect the radius of a region. Region-like features can be referred to as being a ‘region of interest’ or ‘interest region’, for example, blob features [9], Maximally Stable Extremal Region Detector (MSER) [10] and Salient Regions [11]. In some circumstances, interest region detection methods can include additional attributes for regions when considering affine-invariance. Such circumstances need to describe the regions as ellipse regions [12] rather than circular regions. Therefore, we can extend the concept of interest regions from interest points. An interest region is a region that has the following characteristics:

- *It has all the characteristics of an interest point.*
- *Optionally, the notion of interest regions could include a set of attributes to describe the shape of the region. For example, if we want to describe the region as an ellipse shape, we need two additional attributes: the angle of the main axis and the ratio between the two axes.*

Edge-based features, lines and curves can be used as alternative features for applications where the interest points or interest regions methods apply. The earlier edge-based features are successful in applications where the objects are relatively simple [13]. Later developments of these feature detection methods become relatively stable under noise perturbations [5 and 6] and can be used for feature matching [14]. Given the advantages of optimization methods such as generalized Hough voting and dynamic programming, edge-based features or lines can be successfully applied in object detection tasks where the objects are in a turbulent background [15-17].

Unlike interest points or interest regions, the detection methods of edge-based features, lines or curves usually miss the attribute of scale. Most methods in these areas usually aim to improve the detection robustness under noise. During the 1980’s, the research in edge detection was more interested in finding optimal filters. In 1986, Canny [1] presented a famous edge detection method with three criteria: good detection, good localization and low spurious response. Since then, Canny’s edge detection method has become a common image processing technique. In later computer vision development, research has found that multi-scale filtering is even more important for enhancing edge detection results. In 1998, Lindeberg [18] presented an edge detection method with automatic scale selection. Scale selection enables us to include scale for edge features. Conventional line and curve detection methods are based on Hough voting, least square fitting, or other voting schemes. Hough transform (or Hough voting) has been a well-known technique for line and curve detection since the 1990's [7], but one problem is that Hough based techniques are sensitive to noise.

State-of-the-art line detection methods are LSD [5] and EDLines [6]. LSD utilizes gradient direction to form line support regions, and detects lines by region growing followed by rectangular approximation. EDLines uses an edge drawing technique and least square line fitting method to extract line segments. Both LSD and EDLines detect lines efficiently with good false detection control. However, LSD and EDLines do not include a scale attribute

However, since the scale attribute is optional, all line and curve detection methods can potentially generate interest curves. We will give a formal concept of interest curves in the next section.

2.2 Concept

The concept of *interest curves* can be extended from the concept of interest points. An interest curve can be defined as a curve in an image which in general has the following characteristics:

- *It has a clear, preferably mathematically well-founded definition.*
- *It has a set of well-defined positions which can represent the curve in the image space.*
- *Optionally, the interest curve can include more attributes to describe the shape of the curves.*
- *The local image structure around the interest curves is distinctive.*
- *The interest curve is repeatable under various local and global perturbations in the image domain, such as illumination changes.*
- *Optionally, the notion of an interest curve could include an attribute of scale, which makes it repeatable under image transforms including scale and view point changes.*

There are many line detection methods which can detect lines but not curves. We can utilize the concept of interest curves and define a new concept: *interest lines*. An interest line is a line in the image which has the following characteristics:

- *It has all the characteristics of an interest curve.*
- *It has an additional constraint: The curve is straight enough.*

After providing the formal concept of interest curves, it is valuable to distinguish the differences between concepts of interest curves, interest lines, interest points, and interest regions (**Fig. 2.1)**.

In section 2.1, we identified the differences between interest regions and interest points, noting that interest regions have all the attributes that interest points have. Therefore:

- *Interest points can be seen as a subset of interest regions.*

The term *subset* indicates that a feature detection method in a set is able to generate features of a subset when we use additional constraints or estimations within the detection method. For example, we can estimate an interest region as an interest point. The concept of interest lines is inherited from interest curves and it has additional constraints but no additional attributes. Therefore:

- *Interest lines can be seen as a subset of interest curves.*

Both interest curves and interest lines can be estimated and represented as interest points. For example, we can use the center of an interest curve or interest line as an interest point. Therefore:

- *Interest points can be seen as a subset of interest lines and interest curves.*

As we can see in **Fig. 2.1**, interest points are a subset of interest curves, interest lines, and interest regions. It is a part of the intersection area of interest lines and interest regions. There are parts of the intersection area that do not belong to the area of interest points. What does this mean?

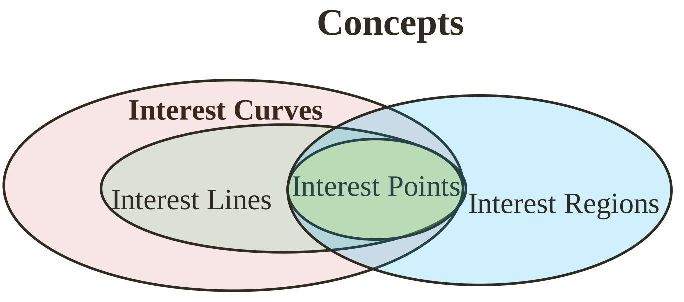


Fig. 2.1 Different concepts used in feature detection methods.

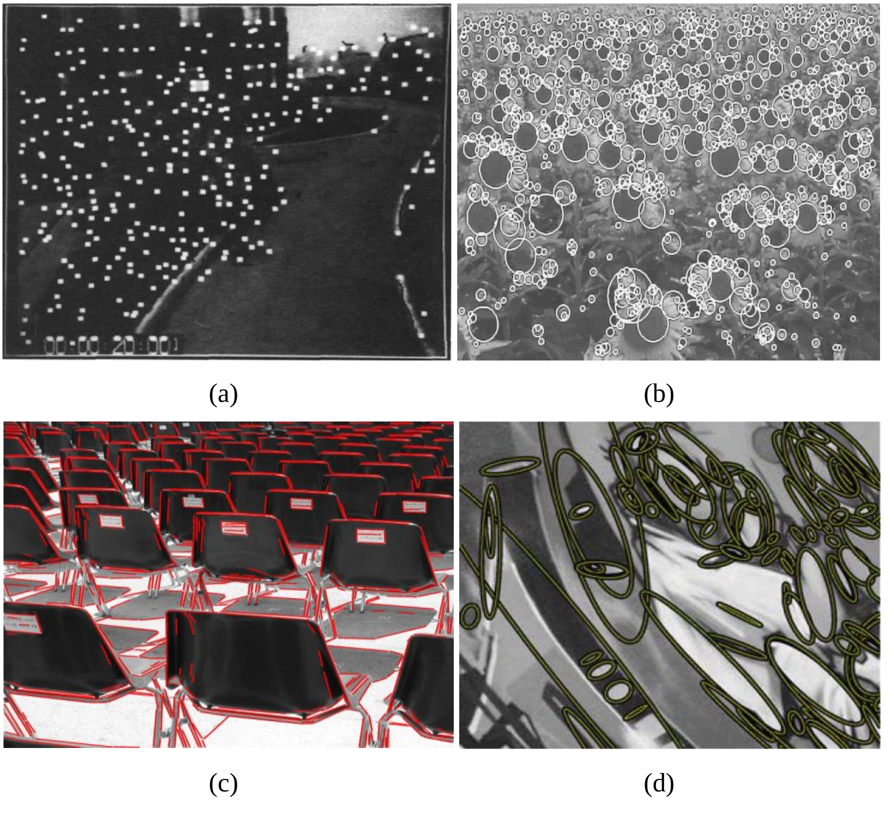


Fig 2.2 Examples of features. (a) Interest points (no scale estimation). Corners detected by Harris detector [3]. (b) Interest points (with scale estimation). Blob features detected by SURF [9]. (c) Interest lines (no scale estimation). Lines detected by EDLine [6]. (d) Interest regions (ellipse estimation). Affine-invariant regions detected by MSER [10, 12].

Interest curves and interest regions can be transformed into each other to some extent and there are many ways to perform this transformation. For example, if we set the scale of the interest curve as one axis of an interest region and the curve length as the other axis of the interest region, we can transform an interest curve to an interest region. On the other hand, we can also transform an interest region into an interest curve. For example, if we detect the border of the region and then detect the curves or lines of the border, then an interest region can be represented as a set of interest curves or lines. However, we should not force transformations between interest regions and interest curves because the method of interest curve detection or interest region detection are usually designed with geometrical meanings.

2.2.1 Examples

When we have clear concepts of different feature detection methods, it is valuable to review and classify some of the well-known feature detection methods. We present some examples of these methods in **Table 2.1** and show some feature examples in **Fig. 2.2**.

Table 2.1 Examples of feature detection methods. Here “no scale” means no proper scale estimation of the features.

Method Name	**Feature Type**	**Brief Description**
Harris Corner [3]	Interest point (no scale)	A corner detector using local auto-correlation function.
FAST [20]	Interest point (no scale)	A machine learning based corner detection method.
SIFT[8]	Interest point (with scale)	Detects extremes of difference of Gaussian in scale-space.
SURF[9]	Interest point (with scale)	Detects blob features using determinant of Hessian in scale-space.

ORB [21]	Interest point (with scale)	Uses the FAST detector in an image pyramid. The scale is not as accurate as in SIFT or SURF.
LSD [5]	Interest line (no scale)	Uses the gradient direction to form line support regions, and detects lines by region growing followed by rectangle approximation.
EDLine [6]	Interest line (no scale)	Uses an edge drawing technique and a least squares line fitting method to extract line segments.
MSER [10]	Interest region (can be ellipse shape)	Detects regions using an “extremal” property of the intensity function in the region and on its outer boundary. It is affine-invariant.
Salient Regions [11]	Interest region (can be ellipse shape)	Detects salient regions in scale-space. It is affine-invariant.

2.3 Data structure

After providing the concepts of interest curves and interest lines, we propose a data structure for these. Interest points have a well-defined data structure in the OpenCV library [22] where all the interest point detectors have been re-programmed and follow the same standard. An interest point (also referred to as keypoint) in OpenCV is presented as:

class **Keypoint**

{

(x, y): *Keypoint position in image*

Size: *Diameter of the meaningful keypoint neighborhood, which is related to the scale estimation.*

Angle: *Computed orientation of the keypoint.*

Response: *Indicating how strong the keypoint is. Can be used for further sorting and subsampling.*

Octave: *Pyramid layer from which the keypoint has been extracted.*

}

According to the concept of interest curves, the difference between an interest curve and an interest point is that an interest curve has a set of well-defined positions which can represent the curve in the image space. Our proposal is that we follow a principle of simplicity and only keep the necessary points of the curve. Our standpoint is that there are three necessary points to describe a curve: one middle point and two endpoints. The middle point may not seem necessary for a line, but we argue that it is necessary since we cannot assume a line is perfectly straight. Our interest curve (can be referred as to keycurve) data structure is:

class **Keycurve**

{

Keypoint M: *The middle point of the curve*

Keypoint L: *"left" endpoint of the curve*

Keypoint R: *"right" endpoint of the curve*

Angle: *Optional. Computed orientation of the keycurve.*

Length: *Optional. Computed length of the keycurve*

Straightness: *Optional. Computed straightness of the keycurve.*

Response: *Optional. Indicating how strong the keycurve is.*

}

In our keycurve data structure, three keypoints follow the OpenCV data structure. The scale of three keypoints can be estimated by the scale of curve. Additional curve information, for example length, straightness, etc., can be utilized for various applications.

2.3.1 Scale estimation

Most existing line or curve detection methods cannot provide scale estimation. However, lines or curves are different from interest points and their length can directly be used as a parameter to estimate the scale for three keypoints of the interest curve. Therefore, the scale of each keypoint is dependent on the design of each specific method. In our implementation of interest curve detection (Chapter 4), we present a scale estimation using the scale-space concept.

2.3.2 Definition of "left" and "right" keypoint

The definition of "left" keypoint and "right" keypoint is important for feature matching and related applications. If the orientation (angle) of the interest curve has already been defined, then the "left"/"right" keypoints can simply be defined as endpoints on the "left"/"right" of the curve direction (**Fig. 2.3**). Without a proper definition, ambiguous "left"/"right" keypoints will affect the correctness of curve matching and related applications.

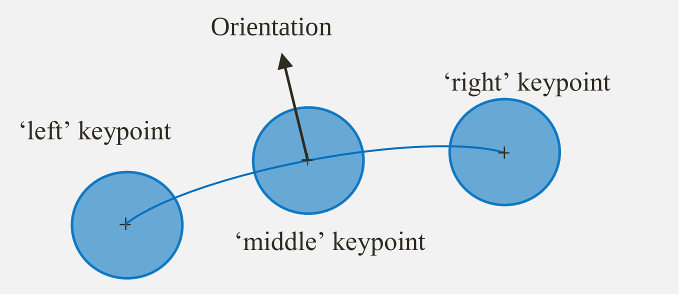


Fig. 2.3 An example of an interest curve.

2.4 Guideline for interest curve detection

In this section, we will analyze some feature detection methods and derive a guideline for interest curve detection methods. These feature detection methods usually include two parts: feature detection and feature description. In the following analysis, we will only discuss the feature detection part.

SIFT: SIFT is a scale-invariant interest point detector. First, given an input image, an image pyramid is generated by applying Gaussian smoothing and down-sampling (**Fig. 2.4**). A pyramid contains ***n*** octaves. Secondly, on each octave image in the pyramid, SIFT generates ***m*+1** octave layers where each layer is a convolution of the octave image with a Gaussian smoothing kernel with various scale parameters. Thirdly, by applying difference of Gaussians (DoG), ***m*** DoG images are generated in each octave. In the next step, SIFT detects the local maxima of DoG in both image and scale space [23]. The next steps include noise cancellation, interpolation, size estimation, etc.

SURF: SURF is also a scale-invariant interest point detector. It also uses these three important concepts: pyramid, octaves and octave layers. It detects blob features using the Determinant of Hessian (DoH). However, the principle of generating an image pyramid is different from SIFT. In SIFT, each octave has a different size but in SURF all octave images have the same size. In SURF, DoH filters in coarser octaves have rather large sizes. Benefited by the design of DoH filter using integral image, large size DoH does not demand additional computational cost.

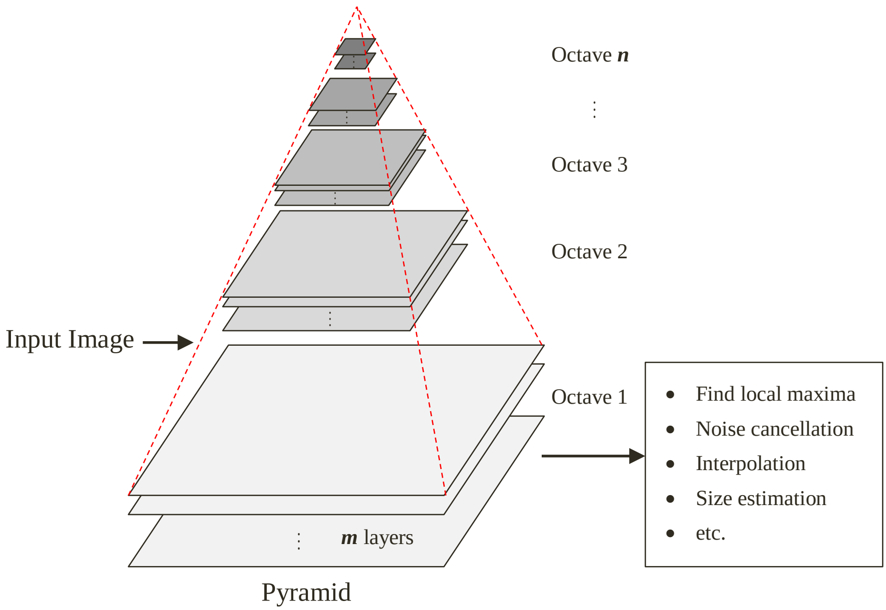


Fig 2.4 Scheme of scale-invariant feature detection.

FAST: FAST uses a machine learning technique to achieve fast corner detection. It is not able to provide scale estimation and hence it is not scale-invariant. In section 4.2.2, we can see that FAST has low *repeatability* under scale change. Here, we can consider the number of octaves ***n*** as 1, and the number of octave layers ***m*** as 1.

ORB: ORB is also a scale-invariant interest point detector. It first generates a pyramid containing ***n*** octave images. On each octave, the FAST detector is applied to detect corners. However, without generating ***m*** octave layers, ORB is not able to provide an accurate scale estimation. In section 4.2.2, we can see that ORB has lower *repeatability* under scale change compared with both SIFT and SURF. Here, we can consider the number of octave layers ***m*** as 1.

EDLine: EDLine uses edge drawing and a least squares line fitting method to extract line segments. It does not use the concept of pyramid and octave layers. Therefore, in section 4.3.2, we can see that EDLine has lower *repeatability* under scale change, compared with SIFT and SURF. Here, we can consider the number of octaves ***n*** as 1, and the number of octave layers ***m*** as 1.

After the observation above, the conclusion can be made that feature detectors with good scale estimation all use image pyramid and more than one octave layers. Therefore, a guideline can be suggested for scale-invariant feature detection, which also applies to scale-invariant interest curve detection:

- *Generate an image pyramid which contains **n** (**n**≥1) octaves.*
- *On each octave, generate **m** (**m**≥1) octave layers.*
- *Detect local maxima in scale-space, constructed by **n** octaves and **m** octave layers.*
- *Apply various feature detection strategies.*

In our implementations of edge detection, corner detection and curve detection, we will follow these guidelines.

3 Evaluation

According to the attributes of interest curves, this chapter proposes a standard evaluation methodology for both curve detectors and curve descriptors.

Recall the attributes of interest points, interest regions, and interest curves: they should be repeatable under various local and global perturbations in the image domain, such as illumination changes; they should be repeatable under image transforms including scale and view point changes.

The Oxford Benchmark provides a tool to evaluate interest point (or region) detectors and descriptors [12, 24]. In the last ten years, it has become a standard evaluation tool for interest point detectors and descriptors. The performance of interest point (or region) detectors and descriptors is measured against changes in viewpoint, scale, illumination, defocus and image compression. Interest curves have similar attributes as interest points and interest regions so we can extend the Oxford Benchmark to measure the performance of interest curve detectors and descriptors.

3.1 The Oxford Benchmark

The Oxford Benchmark evaluation tool consists of three parts: *dataset*, *detector evaluation* [12] and *descriptor evaluation* [24]. The main parameters in the performance measurement are: *overlap error*, *correspondence*, *repeatability* and *1-preceision vs. recall* (**Fig. 3.1**).

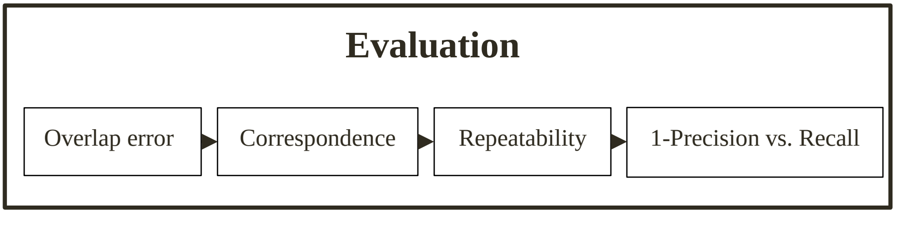


Fig. 3.1 Evaluation parameters.

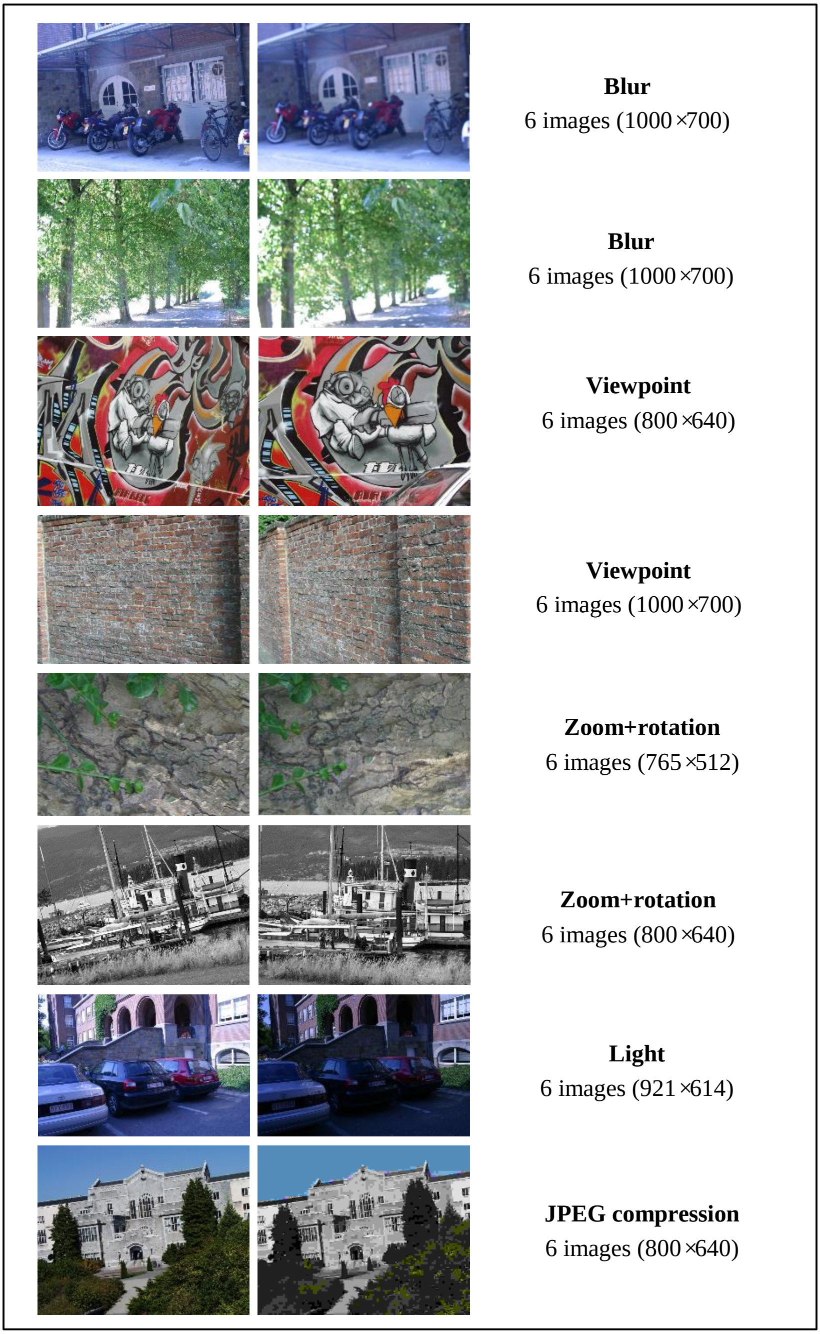


Fig 3.2 Datasets used in the Oxford Benchmark. The first column shows the first images in each category. The second column shows the images in the dataset that represent changes in the images (blur, viewpoint, scale, rotation, light and JPEG compression). The last column labels the type of change and the image resolution.

3.1.1 Dataset

The Oxford Benchmark provides a dataset containing 8 categories. Each category contains 6 images and 5 homographies between the first images to the other 5 images (**Fig. 3.2**). Each category represents one type of change in the images (blur, viewpoint, scale, rotation, light, JPEG compression). For each category, the changes in the six images range from a small to large extent.

3.1.2 Overlap error

The first important measurement parameter is *overlap error*. Two regions are deemed to correspond if their *overlap error* is sufficiently small:

$$Overlap\ error = 1 - \frac{R_{\mu_a} \cap R_{(H^T \mu_b H)}}{R_{\mu_a} \cup R_{(H^T \mu_b H)}} < \epsilon_o \tag{3.1}$$

where R_μ represents the elliptic region and H is the homography relating two images. The union of the regions is $R_{\mu_a} \cup R_{(H^T \mu_b H)}$, and the intersection of the regions is $R_{\mu_a} \cap R_{(H^T \mu_b H)}$. The area of the union and the intersection of the regions are computed numerically. As shown in **Fig. 3.3**, one region is R_{μ_a}, the other region is $R_{(H^T \mu_b H)}$, and the intersection area is shown as the shadow area.

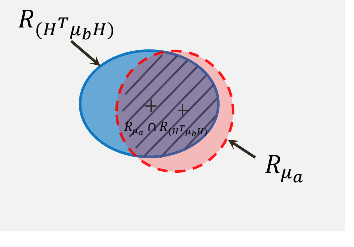


Fig. 3.3 Overlap of two regions. The overlap area is shown as the shadow area.

3.1.3 Correspondence

The number of *correspondences* is counted as the number of regions that have corresponding regions in a projected image using homography. If one region has more than one correspondence, only the best one is counted as a correspondence in this calculation. Only the regions located in the part of the scene present in both images are taken into account.

3.1.4 Repeatability

The *repeatability* score for a given pair of images is calculated as the ratio between the number of correspondences and the smaller number of regions in the image pair:

$$repeatability = \frac{\#\,\mathrm{correspondences}}{\#regions} \tag{3.2}$$

Repeatability and *correspondences* are used for evaluating interest region (point) detectors. A good feature detector should have a high *correspondence* and *repeatability* score.

3.1.5 1-precision vs. recall

1-precision vs. recall is a curve used to evaluate feature descriptors. Two regions, *A* and *B*, are matched if the distance between their descriptors D_A and D_B is below a threshold *t*. If a matched region pair correspond, like described in section 3.1.2, the matching is accepted as a correct match. Otherwise the match is a false match. *Recall* is the number of correctly matched regions with respect to the number of corresponding regions:

$$recall = \frac{\#correct\ matches}{\#correspondences} \tag{3.3}$$

The *1-precision* is the number of false matches respect to the total number of matches.

$$1 - precision = \frac{\#false\ matches}{\#total\ matches} \tag{3.4}$$

By varying the threshold *t*, the *1-precision vs. recall* curve can be obtained. A good feature descriptor must have a high *recall* value and a small *1-precision* value.

3.2 Extending to interest curves

To extend the Oxford benchmark to interest curves, we need to find a way to apply the same measurements to curves: *overlap error, correspondence, repeatability* and *1-precision vs. recall*.

3.2.1 Dataset

Interest curves share similar attributes with interest points and interest regions: they should be repeatable under illumination changes and under image transforms including scale, view point changes. Therefore, the dataset of the Oxford Benchmark is, generally speaking, able to evaluate interest curve detectors and descriptors.

However, some categories of the dataset contain repeated textures of different forms, for example, the "*trees*" and "*wall*" categories cannot generate well-defined edges. Therefore, some categories are not adequate or even fair enough to evaluate edge-based features, including interest curves. In our future work, we will produce an additional dataset as a supplement.

Previous research in edge detection, line detection and curve detection places a large evaluation importance to the stability under noise. Usually, the test images used in previous research contain manually added noise: white noise, pepper noise, or Gaussian noise. In our opinion, the noise in natural images is more objective for evaluation purposes. Therefore, we will not add any additional dataset for stability evaluation under noise.

3.2.2 Overlap error

In Chapter 2, we propose a standard data structure for interest curves. Each interest curve can be represented by three keypoints. The *overlap error* of two interest curves can be estimated by averaging the overlap error of the three keypoints (**Fig. 3.4**):

$$Overlap\ error = \frac{e_l+e_m+e_r}{3} \qquad (3.5)$$

where e_l, e_m, e_r, represent the overlap error of three keypoints in interest curve. A full estimation of the overlap error will be:

$$Overlap\ error = 1 - \left(\frac{R_{\mu_{la}} \cap R_{(H^T \mu_{lb} H)} + R_{\mu_{ma}} \cap R_{(H^T \mu_{mb} H)} + R_{\mu_{ra}} \cap R_{(H^T \mu_{rb} H)}}{R_{\mu_{la}} \cup R_{(H^T \mu_{lb} H)} + R_{\mu_{ma}} \cup R_{(H^T \mu_{mb} H)} + R_{\mu_{ar}} \cup R_{(H^T \mu_{rb} H)}}\right) \tag{3.6}$$

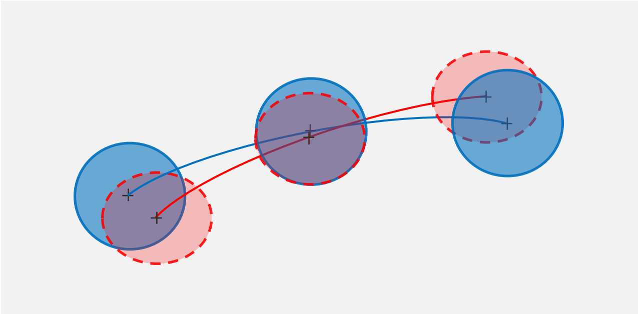


Fig. 3.4 Overlap of two interest curves. Each interest curve can be represented as three keypoints and the overlap error of an interest curve can be measured using the combined overlap error of these three keypoints.

3.2.3 Correspondence, repeatability, 1-precision vs. recall

The *correspondence*, *repeatability* and *1-precision vs. recall* can be calculated using the same equations as for interest points (or regions) using the curve overlap error from the previous section.

3.3 Conclusion

The new evaluation framework for interest curve detectors and descriptors share the same performance parameters as the Oxford Benchmark and therefore, interest curve and interest point methods can be evaluated within the same framework. Using this new evaluation framework, line detection methods can be re-evaluated on the *repeatability* performance. Additionally, this framework can be used for evaluating line and curve descriptors.

4 Implementation

In this chapter, we present our implementation of interest curve detection and description. It includes our contributions in edge detection, corner detection, curve and line detection and description.

We present a unique multi-functional, efficient and robust interest curve detection method (**Fig. 4.1**). Our interest curve detection method unifies the concept of edges, corners, curves and lines. It can output robust multi-features in an efficient way and it utilizes our contributions in various feature detection methodologies.

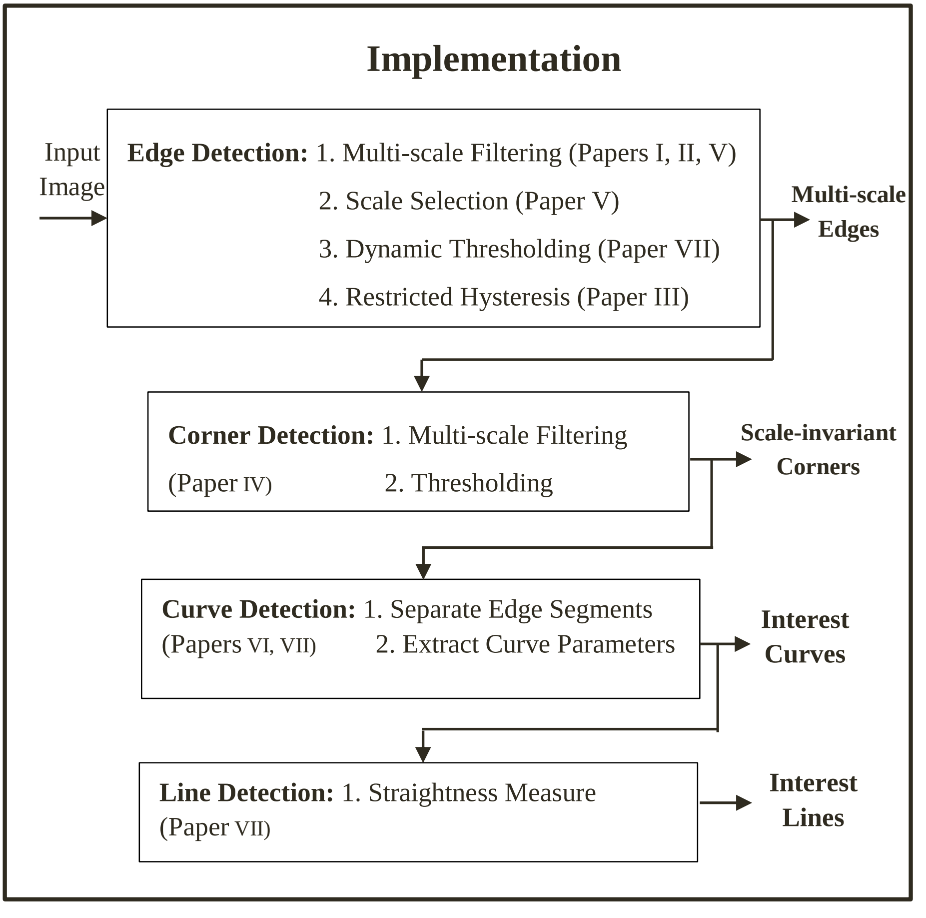


Fig. 4.1 The proposed multi-functional interest curve detector.

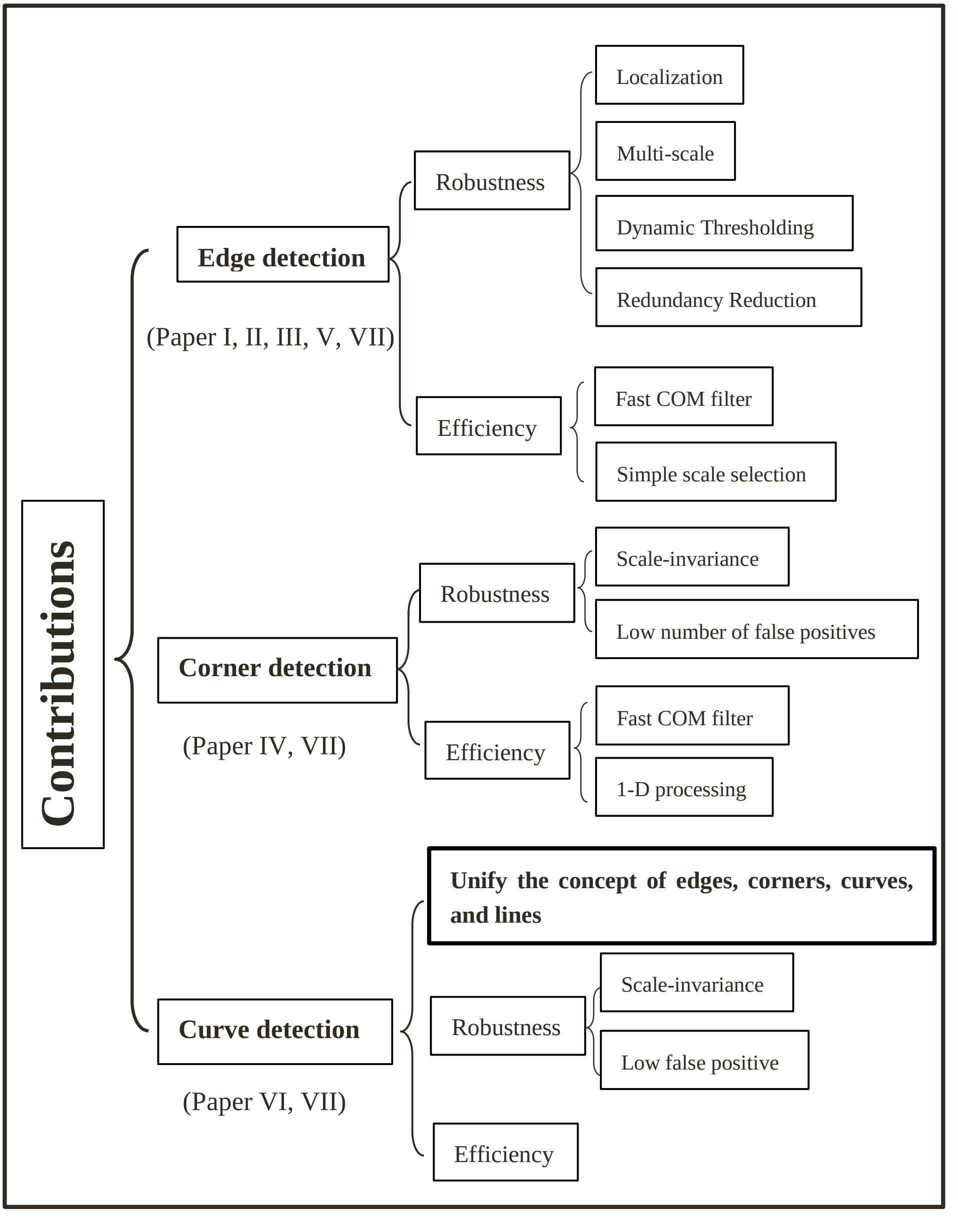


Fig. 4.2 Contributions to feature detection methods.

In our research we have covered the most important attributes for various features in the aspects of robustness and efficiency (**Fig. 4.2**). We will present our contributions step by step in the following sections.

4.1 Edge detection (Paper I, II, III, V, VII)

4.1.1 Good detection and Good localization (Paper I)

In Paper I, edge detection is improved in terms of three research objects: *good detection*, *good localization*, and *only one response to a single edge*.

We propose a *multi-scale edge detection algorithm* based on proportional scale summing and independent thresholding.

Good detection: Our analysis shows that proportional scale summing successfully improves detection by applying independent thresholds on multi-scale gradient images.

Good localization: The proposed method improves edge detection and localization by summing gradient images with a proportional parameter c^n ($c < 1$); which ensures that the detected edges are as close as possible to the position in the fine scale.

Response to a single edge: We employ non-maxima suppression and a thinning step similar to Canny's edge detection framework on the summed gradient images.

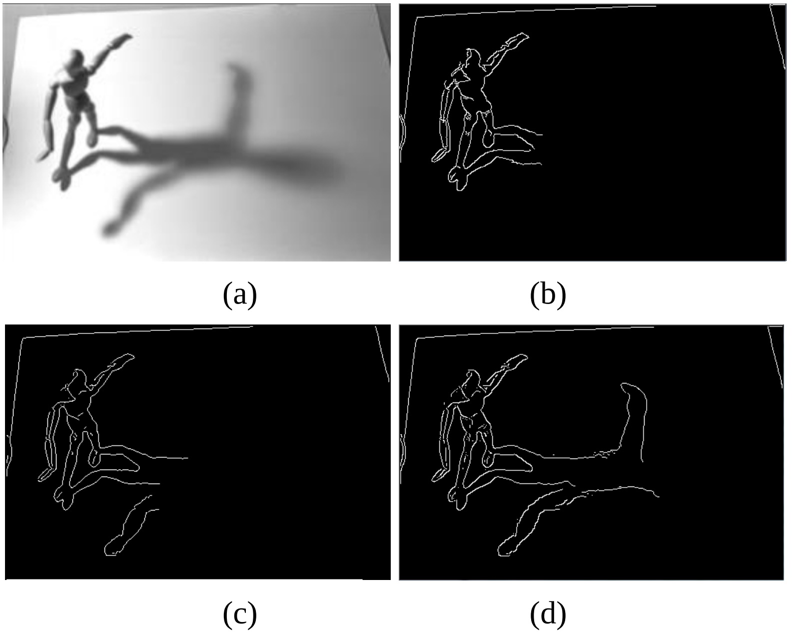


Fig. 4.3 (a) natural images with blurring effect [25]; Edge detection results of (b) Canny; (c) SMED; (d) The proposed PSSED algorithm.

The proposed method leads to better edge detection performance than the Canny edge detector [1] and SMED detector [26].

Our proposed algorithm outperforms other edge detection algorithms, especially when it comes to blurred edges (**Fig. 4.3**).

4.1.2 Fast COM filter (Paper II, V)

In Papers II and V, a new edge detection method was proposed, which computes the image gradient using the concept of Center of Mass (COM). The algorithm runs with a constant number of operations per pixel, independently from its scale by using an integral image. Compared to conventional convolutional edge filters such as the Sobel edge filter, the proposed method performs faster when the region size is larger than 3×3 pixels. The proposed method can be used as a framework for multi-scale edge detectors when the goal is to achieve fast performance. Experimental results show that edge detection by COM is comparable to Sobel filter (**Fig. 4.4**).

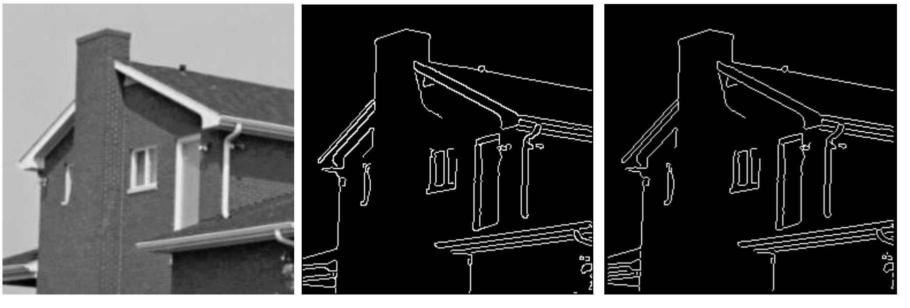


Fig. 4.4. Left: Input image. Middle: Edge detected using Sobel filter. Right: Edge detected using COM filter.

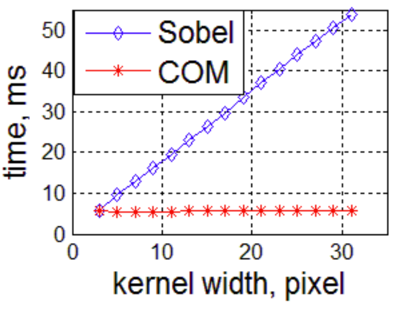


Fig. 4.5 Speed comparison.

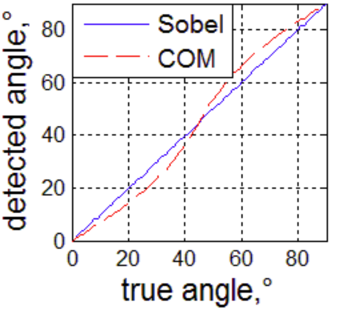


Fig. 4.6 Angle accuracy.

As shown in **Fig. 4.5**, the proposed algorithm consumes approximately 5.7 *ms* invariant to filter size. Sobel filter consumes 6 *ms* on filter size of 3×3 and 54 *ms* on filter size of 31×31.

Unlike the Sobel filter, which has Gaussian built-in smoothing, the COM filter is totally square. Therefore, the angle accuracy calculated by COM will not be as accurate as for the Sobel filter **(Fig. 4.6**). We test the angle accuracy by applying both algorithms on a set of synthetic images. The maximum angle error of the COM filter is 6.62 degrees. This small error is acceptable in edge detection operations.

4.1.3 Redundancy reduction (Paper III)

In Paper III, unnecessary redundancy of detected edges are eliminated.

In edge detection algorithms, there is a common redundancy problem, especially when the gradient direction is close to -135°, -45°, 45° and 135°. The double edge effect appears on the edges around these directions, caused by the discrete calculation of non-maximum suppression. Many algorithms use edge points in further tasks, such as line extraction, curve detection, matching and recognition. Therefore it is important to remove redundant edge points.

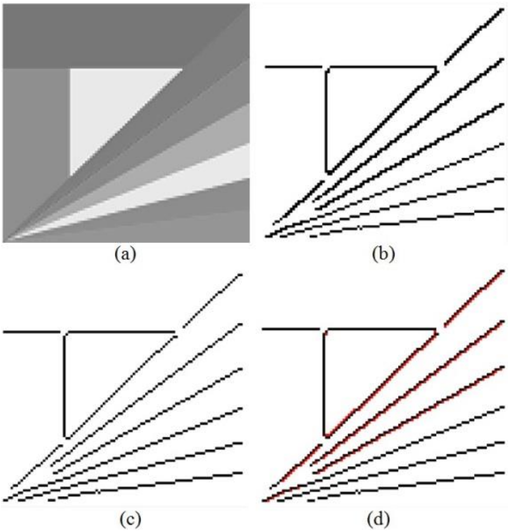


Fig 4.7 (a) Synthetic image; (b) Edge detected by Canny edge detector; (c) Edge detected using restricted hysteresis; (d) Shows a comparison between (b) and (c), red pixels are the edge points removed using our method.

Redundancy is a very important factor for algorithm speed and accuracy. We found that most edge detection algorithms have a redundancy of 50% in the worst case (and 0% in the best case), depending on the edge direction distribution. The common redundancy rate on natural images is approximately between 15% and 20%. Based on Canny's framework, we propose a restriction in the hysteresis step. Our experiment shows that the proposed restricted hysteresis reduce the redundancy successfully. In **Fig 4.7**, we can see that when the edge direction is close to ±45°, the redundancy is about 50% and it reduces when the edges become more and more horizontal.

Fig. 4.8 First row: input images with illumination change. Second row: detected edges without using DT. Third row: detected edges using DT.

4.1.4 Illumination robustness (Paper VII)

Edge detection algorithms need thresholds to filter noise. However, choosing the proper threshold to trade-off illumination change and signal to noise ratio is always difficult and seldom satisfactory. To overcome this problem, Paper VII proposes a dynamic threshold (DT) method:

$$h' = \frac{h\,\bar{\bar{G}}}{b} \tag{4.1}$$

where h is the original user defined threshold, $\bar{G}$ is the average gradient along all gradient extremes generated from the non-maximum suppression step, and b is a constant parameter which enables h' in a preferred magnitude. **Fig. 4.8** shows that the dynamic threshold significantly improves the edge robustness under illumination change. The reason we use $\bar{G}$ instead of the average image illumination is that $\bar{G}$ also takes the signal to noise ratio into account. For example, if a dark image contains a high level of pepper noise, equation 4.1 will generate a high threshold to suppress the pepper noise.

4.1.5 Scale selection (Paper V)

Scale selection is a good method to generate robust edges. Additionally, a scale selection method can generate scale information which can benefit interest curve detection. By simplifying Linderberg's scale selection concept [18] and extending Canny's concept, the proposed multi-scale scheme is composed as follows:

1. Filter the image using various sizes of Sobel or COM filters (here we apply 5 filter sizes: 3×3, 5×5, 9×9, 17×17 and 31×31).
2. Scale the gradient maps by multiplying with *S*. *S* decreases when the scale goes from fine to coarse. This means that the finer scales are favored.
3. Apply 3D NMS on the gradient maps. Firstly, find 2D maxima along the gradient direction. Then eliminate the maxima points that are no larger than their neighbor pixels in finer and coarser scales.
4. Apply 3D hysteresis. Similar to Canny's hysteresis method, extended 3D hysteresis applies the same double thresholding strategy to filter out weak edges which are not connected to strong edges.

Fig. 4.9 (e) and **Fig. 4.9 (f)** show the resulting edge maps using the proposed multi-scale scheme using the Sobel and COM filters utilizing the same threshold. Both figures show that the proposed multi-scale edge detection scheme locates blurred edges well without being affected by noise.

Table 4.1 shows the time consumption of the proposed scale selection schemes using both the Sobel and COM filters. The total time consumption using the COM filter is half the time using the Sobel filter. The proposed method is fast enough to be used in practical applications.

Table 4.1 Time consumption of the proposed scale selection scheme using the Sobel and COM filters (using the 512×512 'Lena' image).

Time Consumption	Gradient calculation	Scaling, 3D NMS, 3D hysteresis, etc.	Total
Using Sobel	92 ms	39 ms	131 ms
Using COM	28 ms	39 ms	67 ms

(a) *(b)* *(c)*

(d) *(e)* *(f)*

Fig. 4.9 Edge detection results. (a) 'Lena' image. (b) Using 3×3 Sobel filter and Canny's scheme with high thresholds. (c) Using 3×3 Sobel filter and Canny's scheme with low thresholds. (d) Using 3×3 COM filter and Canny's scheme with high thresholds. (e) The proposed multi-scale method using the Sobel filter. (f) The proposed multi-scale method using the COM filer.

4.2 Scale-invariant corners (Paper IV)

In Paper IV, a fast scale-invariant corner keypoint (SICK) detection method is proposed. SICK considers both robustness and efficiency of the algorithm and is more robust than other state-of-the-art algorithms during evaluation using the Oxford Benchmark.

4.2.1 Method

Firstly, an image pyramid is generated by Gaussian smoothing and down-sampling. Secondly, edge chains are extracted by applying a fast scale selection method based on the COM filter for each octave of the pyramid. Thirdly, edge based corner detection is implemented by detecting the edge direction change.

We measure the corner score by calculating the magnitude of gradient direction change by following one-dimensional edge chains. The corner score is calculated as:

$$c = \frac{\left(\sum_{t-w/2}^{t} G_x - \sum_{t}^{t+w/2} G_x\right)^2 + \left(\sum_{t-w/2}^{t} G_y - \sum_{t}^{t+w/2} G_y\right)^2}{\left(\sum_{t-w/2}^{t} G_x\right)^2 + \left(\sum_{t-w/2}^{t} G_y\right)^2 + \left(\sum_{t}^{t+w/2} G_x\right)^2 + \left(\sum_{t}^{t+w/2} G_y\right)^2} \tag{4.2}$$

where G_x and G_y are directional gradients, *t* is the measuring location and *w* is the width of the corner filter. In SICK, *w* is determined by the edge scale of each edge pixel.

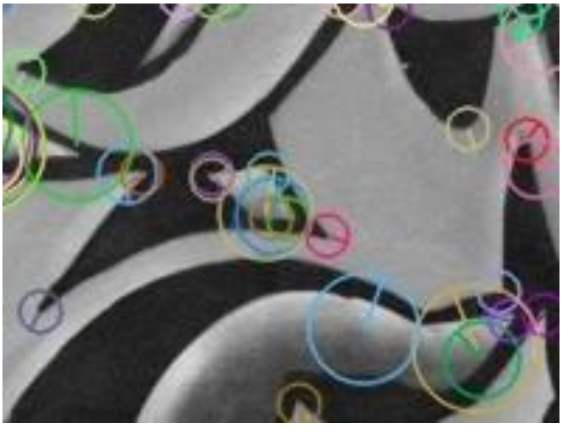


Fig. 4.10. Detected keypoints on the *graf* image using SICK.

Our algorithm has two advantages. Firstly, an edge based corner measure is more efficient than a corner measure using a Harris or Hessian matrix. Secondly, feature descriptors usually need an orientation assignment step to generate keypoint orientation. Since we use the gradient direction as our keypoint orientation, the orientation can be assigned directly without any additional cost.

Fig. 4.10 shows an example of keypoints detected using SICK. **Fig. 4.11** shows an example of keypoint matching using SICK.

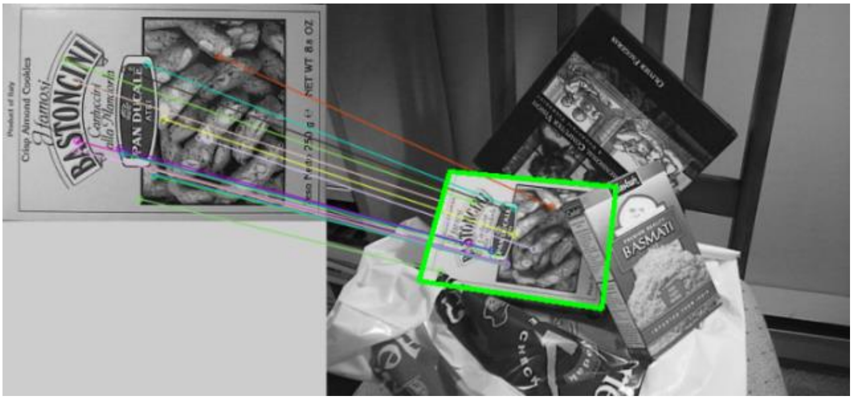


Fig. 4.11. Matching test on planar images using SICK.

4.2.2 Evaluation

We have implemented SICK using 8 filter layers (SICK8) as well as SICK using 3 filter layers (SICK3). SICK8 is implemented in order to achieve the best performance and SICK3 is implemented to test its speed and performance limitations. **Fig. 4.12** illustrates the evaluation results of various detectors using the Oxford Benchmark. SICK outperforms other methods under scale changes, viewpoint changes and JPEG compression. SICK outperforms the other methods under scale change since SICK is relatively unaffected by changes in scale while the performance of FAST, ORB, and BRISK dramatically decreases when the scale increases. SICK's performance under blur change lies somewhere between SIFT's and SURF's performances. SICK's performance under light change is slightly weaker than other methods but still within an acceptable range, similar to SIFT. This result is affected by the edge threshold set in our algorithm and can be improved using dynamic thresholding.

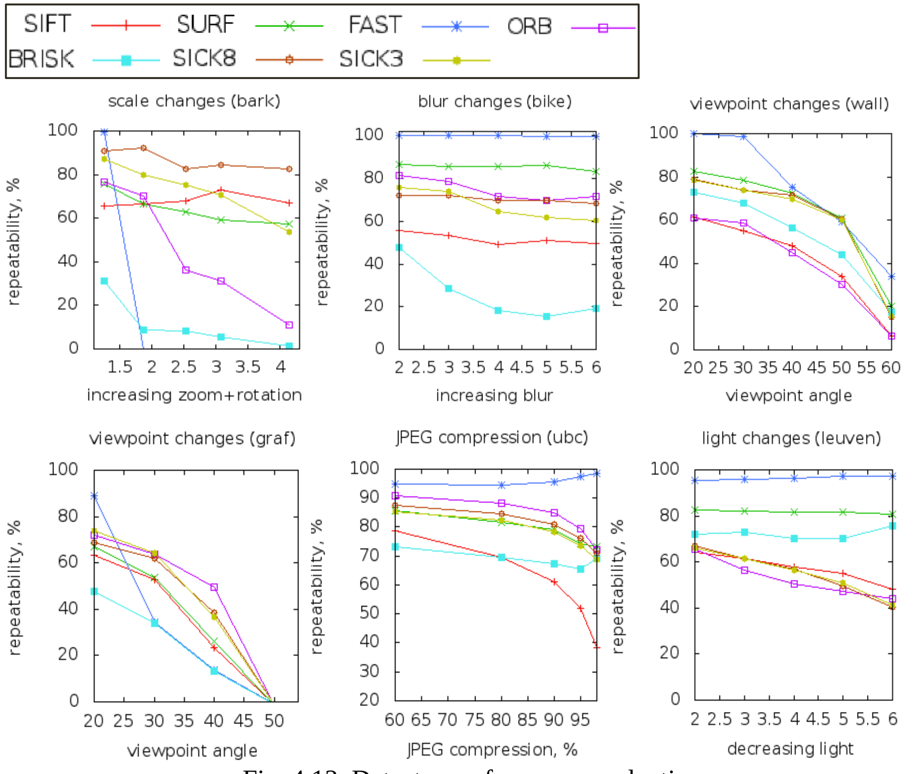


Fig. 4.12**.** Detector performance evaluation.

Table 4.2 shows the time consumption of keypoint detectors. All algorithms are running on an Intel Duo core (4 threads) 2.7 GHZ CPU using a single core single thread. SICK3 (using 3 layers in edge detection) is 34 times faster than SIFT [8] and 5.5 time faster than SURF [9] and slightly slower than ORB [21].

Table 4.2. Time consumption of keypoint detectors.

Detector	SIFT	SURF	ORB	BRISK	SICK8	SICK3
Time(ms)	5024	807	74	180	510	147

4.3 Distinctive curves (Paper VI, VII)

Paper VII presents a state-of-the-art interest curve detector (Distinctive Curves-DICU) that unifies the detection of edges, corners, lines and curves. DICU has three advanced properties: multi-function, efficiency and stability. Multi-function and efficiency are benefited by the inherent property of the DICU work flow. DICU detects edges, corners, curves and lines step by step. Therefore, DICU reduces data dimensionality gradually and requires less and less computation in each step. The principle that ensures the success of each detection step is that we fully utilize the common properties shared by edges, corners, curves and lines in each step. The stability is achieved by applying a scale-space concept in the edge and corner detection steps.

DICU is very stable under various image transformations. After the scale-invariant corners are detected, edges are separated at corner locations and the separated edge segments are represented as DICU features.

We extend DICU to detect distinctive line features (DILine) in scale-space. We simply add a post-validation stage that deletes the curves that do not satisfy a straightness measure.

DICU and DILine are beneficial to a wide range of computer vision applications. Distinctive curves and lines inherently contain richer information than interest point features, therefore they are more robust. We test DICU for a curve matching task where we apply the FREAK [28] descriptor on DICU features. The Oxford Benchmark is used to evaluate the *repeatability* and *1-precision recall* of DICU under these image transformations. DICU outperforms state-of-the-art interest point detection methods (SIFT, SURF, BRISK [27], and ORB), as well as line detectors (EDLines [5] and LSD [6]). DICU's unique representation improves the robustness of feature description and therefore outperforms state-of-the-art keypoint descriptors (SIFT, SURF, BRISK, ORB and FREAK) in our feature matching experiments.

4.3.1 Detection

The DICU features are detected stage by stage. At the same time edges, corner, curves and lines are generated gradually. DICU detection is performed as follows:

1. Generate N octave images using a scale-space pyramid by down-sampling the original image with a set of scale factors (adding optional Gaussian blurring).
2. Detect edges using multi-scale filtering and a scale selection method in each octave of the pyramid.
3. Detect SICK++ features and separate edge segments.
4. Represent the edge segments as DICU features.
5. Select straight curves as lines and represent them as DILine features.

The following paragraphs explain the details of the detection stages. The image pyramid generation follows the same principle as for the interest point detector. Therefore, we do not explain this stage as a standalone section.

Edge detection: The edge detection method is similar to the *scale selection* method shown in Papers V and IV. Furthermore, the previously developed *COM filter*, *restricted hysteresis*, *dynamic thresholding* techniques are merged into the edge detection stage.

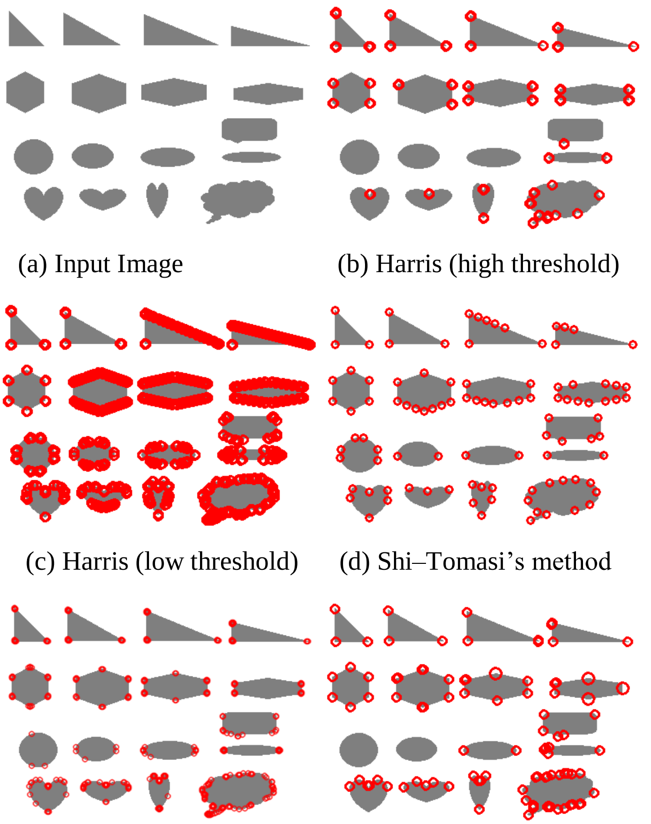
Scale-invariant corner: In Paper VII, the scale-invariant corner detection method (SICK++) is the extended version of the previously developed SICK corner detector [73]. The corner score is measured by calculating the magnitude of gradient direction change by following one-dimensional edge chains. The *corner score* is calculated as:

$$c = \frac{\left(\sum_{t-w/2}^{t} G_x - \sum_{t}^{t+w/2} G_x\right)^2 + \left(\sum_{t-w/2}^{t} G_y - \sum_{t}^{t+w/2} G_y\right)^2}{\left(\sum_{t-w/2}^{t} G_x\right)^2 + \left(\sum_{t-w/2}^{t} G_y\right)^2 + \left(\sum_{t}^{t+w/2} G_x\right)^2 + \left(\sum_{t}^{t+w/2} G_y\right)^2} \tag{4.3}$$

where, G_x and G_y are directional gradients, *t* is the measuring location in the edge chain and *w* is the width of the corner filter. In SICK, *w* is determined by the edge scale of each edge pixel. In SICK++, corner filtering is applied in scale-space and it is applied *M* iterations with multi-size of $w_{[s]}$. In the first iteration, the corner filter is applied using the largest width $w_{[s]}$ and the second largest width $w_{[s-1]}$. The corners are selected as the local maxima which satisfy the following criteria:

$$\begin{aligned}
&1.\ c_{[t,s]} > threshold \\
&2.\ c_{[t-1,s]} < c_{[t,s]}\ and\ c_{[t,s]} \geq c_{[t+1,s]} \\
&3.\ c_{[t,s-1]} < c_{[t,s]}\ and\ c_{[t,s]} \geq c_{[t,s+1]}\ \ if\ s \neq 0\ or\ max \\
&\quad c_{[t,s-1]} < c_{[t,s]}\ \ if\ s = max \\
&\quad c_{[t,s]} \geq c_{[t,s+1]}\ \ if\ s = 0
\end{aligned} \tag{4.4}$$

where *s* indicates scale. The edge segments are then separated at the corner locations. In the next iteration, corner detection is repeated using a smaller *w*. This procedure assures that the unsharp corners can be detected at accurate locations, while producing few false detections.



(a) Input Image (b) Harris (high threshold)

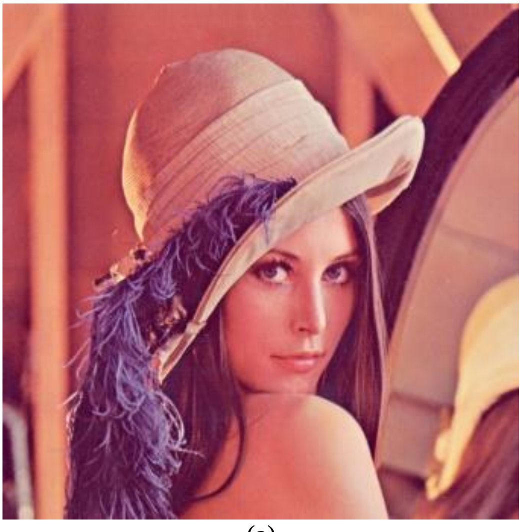
(c) Harris (low threshold) (d) Shi–Tomasi's method

(e) FAST (f) SICK++ (first octave)

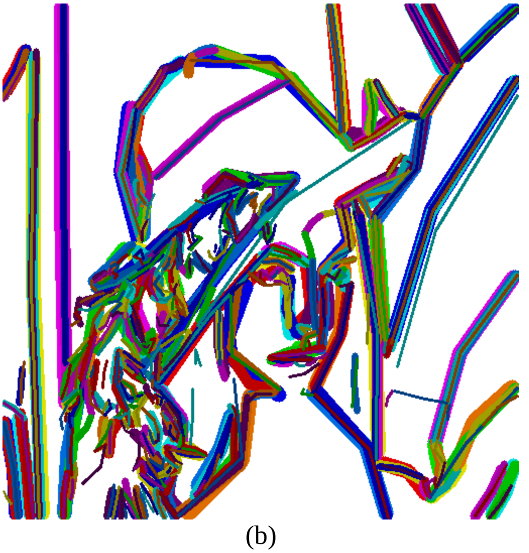
Fig.4.13: Corner detection results using various methods.

Fig.4.13 shows a comparison between various corner detection methods. Harris's method [3] is not capable of detecting unsharp corners (**Fig.4.13 b**). When setting a low threshold, Harris's method detects unsharp corners but it also produces many false detections due to noise. On the contrary, when setting a high threshold, Harris's method would miss detection of most unsharp corners. Shi–Tomasi's method [4] also produces false detections and while FAST [20] produces a better result, it still produces false detections around the true corners. SICK++ produces the best result and is good at detecting unsharp corners while having a low response to noise.

Distinctive curves: The generation of DICU features is very efficient. Each edge chain is separated at corner positions to produce edge segments and they represent smooth curvature changes. These edge segments are represented as DICU features using three keypoints. We have also calculated the additional parameters of DICU including *orientation* and *size* of each keypoint as well as curve *response*, *length* and *straightness*. **Fig. 4.14** shows an example of curve detection using 4 octaves.



(a)

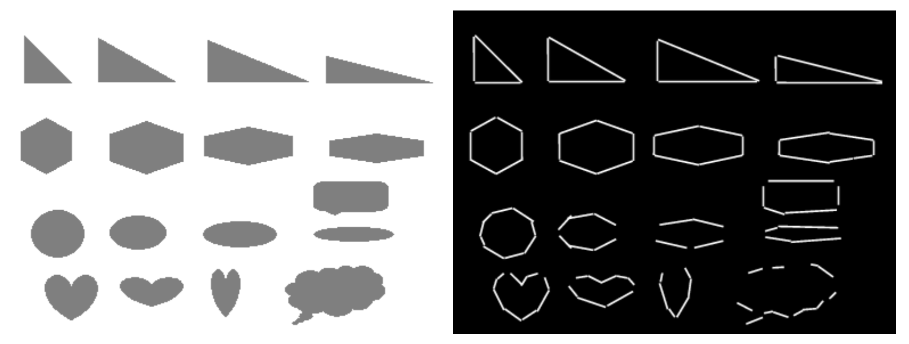


(b)

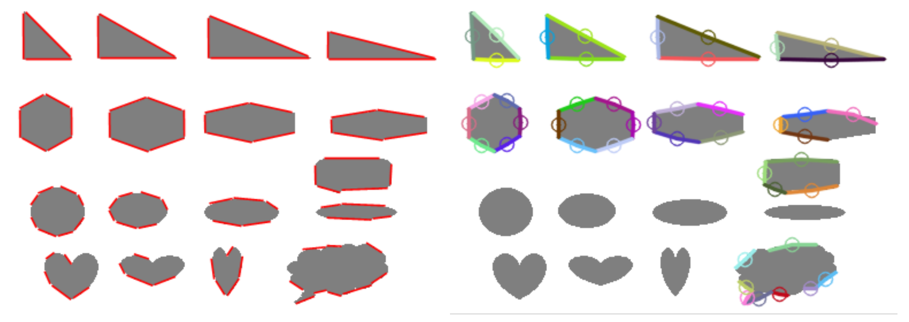
Fig.4.14: (a): input image. (b): Detected curves using 4 octaves. Curves detected in a smaller octave images are represented using a thicker line width. Each curve is represented by two lines connecting the middle point and the two endpoints.

Distinctive lines: To adapt DICU to the detection of distinctive line features, a simple and straightforward way is to keep the curves that are straight enough.

Fig. 4.15 shows a comparison between EDLine [6], LSD [5], and DILine. DILine has very low response to arcs while other methods detect arcs as several short line segments. This is one of the reasons that they are not robust enough under image transformations.



(a) Input Image (b) LSD

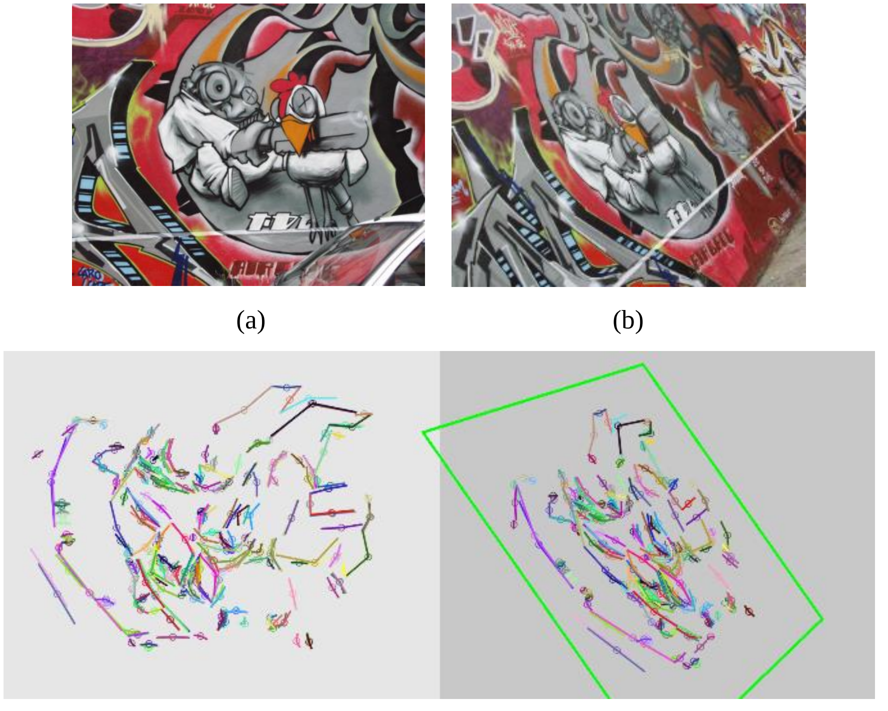


(c) EDLine (b) DILine

Fig.4.15: Line detection results using various methods.

4.3.1 Curve matching

A DICU descriptor as well as a matching method can follow the same principle as for interest point descriptors. A simple way of building curve descriptors is to describe the three keypoints of an interest curve and form a descriptor with a higher dimension. **Fig. 4.16** shows a curve matching example using the FREAK descriptors based on three DICU keypoints (DICU+FREAK).



(a) (b)

(c)

Fig. 4.16**:** Curve matching. (a) Graffiti-1 image. (b) Graffiti-4 image. (c) Matched curves. Each curve is represented by two lines connecting the middle point and the two endpoints. Middle points are labeled as circles. The green box shows where the left image is located in the right image.

4.3.2 Evaluation

In Chapter 3, the evaluation method for interest curve detection and description has been proposed.

Firstly, the *repeatability* of DICU is compared with keypoint detectors (SIFT, SURF, BRISK, and ORB) and line detectors (LSD and EDLines). Experimental results (**Fig. 4.17**) show that DICU is more robust than the other methods under various image transformations. LSD and EDLines do not provide reasonable line scale estimation. Therefore, we use the line length as the factor of keypoint size (*size = line length/2*). Here we also generate three keypoints from each line using the same principle. All the compared algorithms are from the OpenCV [22] library. In OpenCV, LSD and EDLines have been extended using image pyramids.

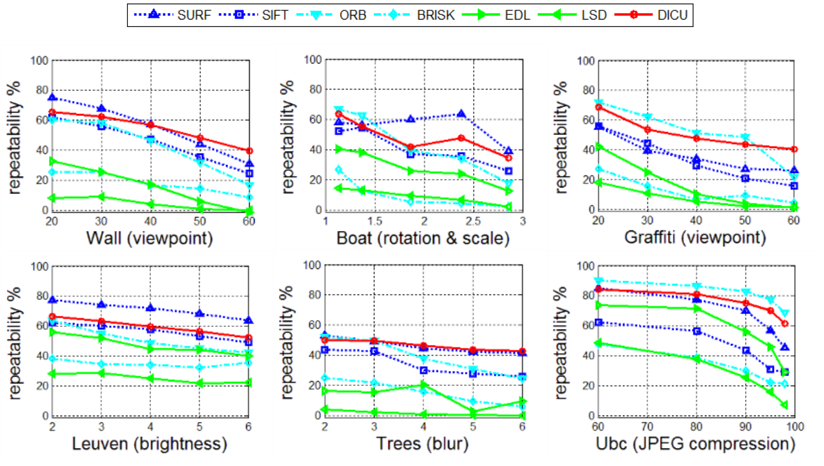


Fig. 4.17 *Repeatability* evaluation under various image transformations using the Oxford Benchmark [12].

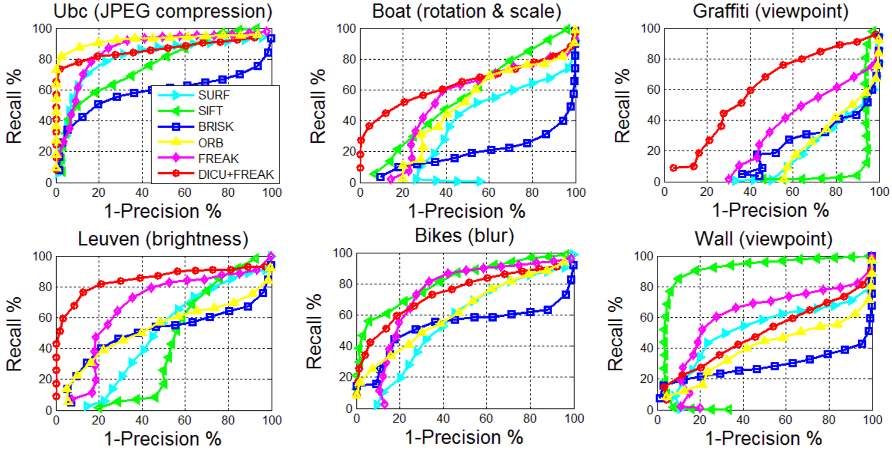


Fig. 4.18 The *recall vs. 1-precision* graph under various image transformations using the Oxford Benchmark [24].

Secondly, the performance of the DICU+FREAK descriptor is evaluated using *1-precision* vs. *recall* graphs. In the evaluation, each method uses its own detector, except for FREAK, which uses the SURF detector. The evaluation shows that DICU+FREAK outperforms the other keypoint descriptors (**Fig. 4.18**).

The run time comparison (**Table 4.3**) shows that DICU is very efficient compared to keypoint detectors. After the edge detection stage, corners, curves, and lines can be detected using around 5% of the total time consumption.

Table 4.3. Average run time comparison of keypoint detection and DICU detection on Oxford datasets. DICU uses 4 octaves, and each octave uses 4 layers.

SIFT	**SURF**	**DICU**	**DICU**
		Edge using Sobel filter: 1.079 s	Edge using COM filter: 0. 531 s
		SICK++: 55 ms	SICK++: 55 ms
		DICU: 7 ms	DICU: 7 ms
		DILine: 1 ms	DILine: 1 ms
6.146 s	**1.079 s**	**Total: 1.133 s**	**Total: 0.596 s**

5 Applications

Image features play important roles in several computer vision applications, such as image registration, 3D reconstruction, object detection and video understanding (**Fig. 5.1**). This chapter discusses the applications which can be benefited from interest curves. More specifically, we present our novel curve-based object detection application framework. Our curve-based object detection framework presents unique properties regarding to efficiency and rotation-invariance.

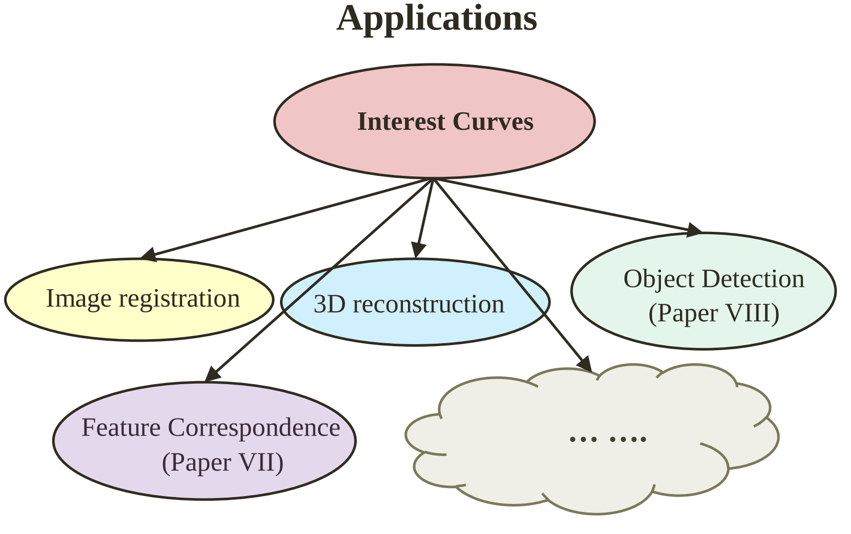
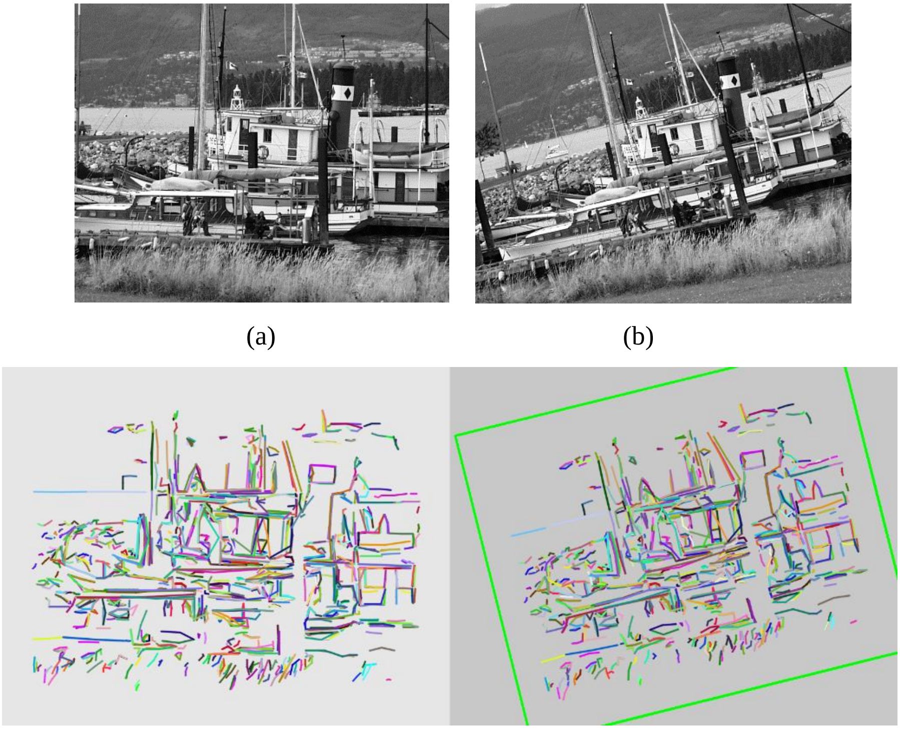


Fig. 5.1 Application areas of interest curves.

5.1 Feature correspondence (Paper VII)

As seen in section 4.3, interest curves can be used for feature correspondence. Solving features correspondence successfully is a key step in homography estimation (**Fig 5.2**), image stitching, 3D reconstruction, etc. Our DICU detector has proved to be more stable than interest point detectors for this kind of task.



(a) (b)

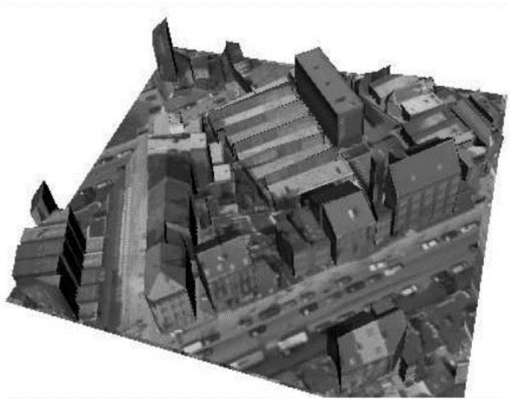
(c)

Fig. 5.2 Curve matching and homography estimation. (a) Boat-1 image. (b) Boar-2 image. (c) Matched curves. Each curve is represented by two lines connecting the middle point and the two endpoints. The green box is the estimated homography showing where the left image is located in the right image.

5.2 3D Reconstruction

Interest curves are useful for 3D reconstruction applications and Caroline *et al.* [29] present a 3D building construction method based on automatic line matching (**Fig. 5.3**). 3D construction requires good line detection and accurate matching results. As seen in section 4.3, conventional line detection methods are inherently unstable under scale and view changes. Scale-invariant interest curve and line detection methods enhance the 3D reconstruction performance since the *repeatability* is increased.

(a)



(b)

Fig. 5.3 Example of 3D construction of buildings [29]. (a) Matched line segments of three different views of the same buildings. (b) Reconstructed 3D model of the scene.

Fig. 5.4. Curve chain matching without local appearance. Left: Input image. Right: Target image and matched curves.

5.3 Object detection (Paper VIII)

Curve features contain richer image geometry information than point features. We have designed a novel object detection algorithm that only utilizes the curve geometries without using further local image appearance (descriptors). We organize an image object as a curve chain where the chain describes curve neighbor relations including relative locations, angles, sizes, and gradient differences. To detect an object, we search for this curve chain in a target image using dynamic programming (DP) [30-32]. **Fig. 5.4** shows an example of the curve chain matching algorithm. It can be seen that most curves are matched on the target.

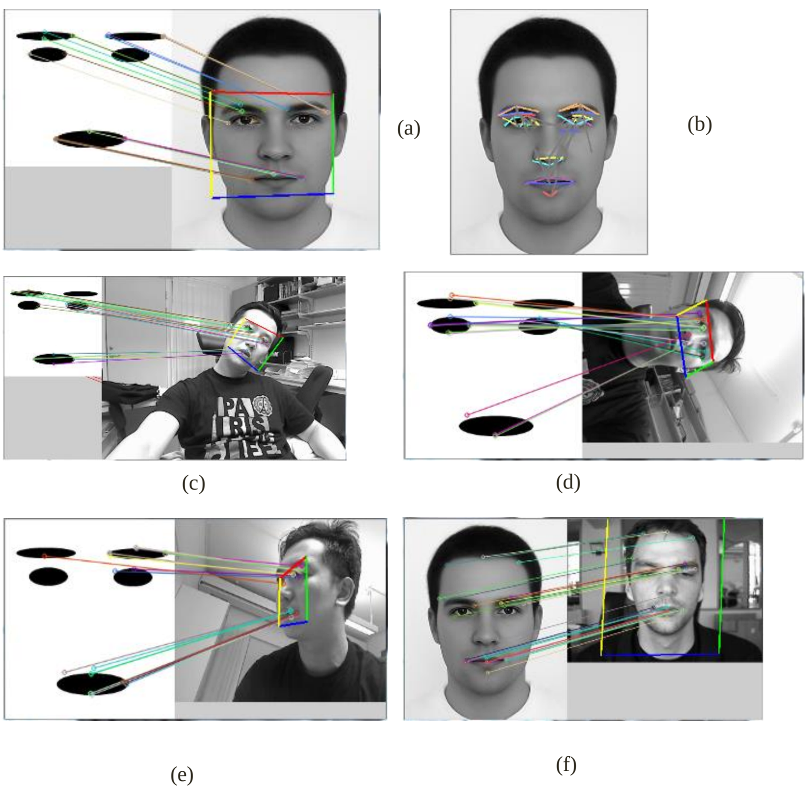


Fig. 5.5. Face detection and homography estimation. (a) Using a simple sketch template to match an average face image [33]. (b) Matched curves by DP. (c-e) matching with rotation change and face pose change. (f) Using a face image as the template.

Curve chain matching is scale and rotation-invariant and also robust to image deformations and this property can solve the problem with rotation in object detection applications. Most object detection algorithms cannot achieve full rotation-invariance in a reasonable computation time. For example, Viola and Jones's face detection algorithm [34] only tolerate small in-plane rotation.

We have evaluated the curve chain matching algorithm for face detection using a simple sketch template **(Fig. 5.5**). Experiments show that our algorithm works well under face rotation and pose changes. Additionally, curve chain matching works well to match a similar object (**Fig. 5.5 f**). Currently, the method does not work when the face is very small within the image, since the curves on the face cannot be detected well enough when the resolution is low. Therefore, it is more suitable for face tracking when there are enough details in the face. In our experiment, face detection using curve chain matching is very efficient: after curve detection, the curve chain matching only consumes 34ms (**Table 5.1**).

Table 5.1. Run time analysis of face detection (test on 300*300 images). Our experiment is conducted on a dual core CPU (3.06 GHz each) with 25% CPU occupation.

	Curve Detection	Matching	Total
Consumption	220 ms	34 ms	254 ms

CHAPTER 5

6 Concluding Remarks

The detection of edges, corners, lines, and curves is a fundamental computer vision problem. However, the research in line and curve detection is still fragmented. This thesis presents our contributions in the area of feature detection methods and related applications. Our contributions range from concept development, evaluation, and implementation to applications.

Firstly, a novel interest curve concept is proposed. Interest curve is a concept derived and extended from interest points. Interest curves are important lines and arcs in an image that are repeatable under various image transformations. Interest curves have clear guidelines and structures for future curve and line detection algorithms and related applications.

Secondly, this thesis presents an evaluation framework for interest curve detectors and descriptors. Our evaluation framework has been developed by extending the Oxford Benchmark, which is a standard method for evaluating interest point (or region) detectors and descriptors.

Thirdly, this thesis presents an interest curve detection and description method: DICU. DICU is a comprehensive work that includes most of our contributions in the area of edge detection, scale-invariant corner detection, curve detection and line detection. Our contributions include the most important attributes required for these features with respect to robustness and efficiency. Beyond these contributions, DICU unifies the concept of edges, corners, lines and curves. Such unification makes DICU a multi-functional, efficient and robust feature detection algorithm. Our contributions in these areas outperform the state-of-the-art methodologies.

Image features have important roles in many computer vision applications, such as image registration, 3D reconstruction, object detection and video understanding.

In fact, many recently developed computer vision applications still use basic edge, corner, line or curve detection methods. Therefore, our research results contribute significant value when it comes to providing a clear and practical framework that can be used as guidance regarding these features.

Curve features contain richer image geometry information than interest points. We utilize this advantage and design a novel curve chain matching algorithm which can benefit a wide range of object detection applications. Our curve chain matching solution has a unique advantage in that it can detect objects in a scale and rotation-invariant manner. Compared with graph matching [35] or Hough voting [15] based object detection methods, curve chain matching achieves a very high efficiency.

In our future work, we will investigate the usage of interest curve features for various computer vision applications such as object recognition. We believe interest curves will become a popular and an invaluable tool for computer vision and image processing applications.

Bibliography

Bibliography

[1] Canny, John. "A computational approach to edge detection." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 6 (1986): 679-698.

[2] Arbelaez, Pablo, et al. "Contour detection and hierarchical image segmentation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33.5 (2011): 898-916.

[3] Harris, Chris, and Mike Stephens. "A combined corner and edge detector." *Alvey vision conference*. Vol. 15. 1988.

[4] Tomasi, Carlo, and Jianbo Shi. "Good features to track." *CVPR94* 600 (1994): 593-593.

[5] Grompone von Gioi, Rafael, et al. "LSD: a line segment detector." *Image Processing On Line* 2 (2012): 35-55.

[6] Akinlar, Cuneyt, and Cihan Topal. "EDLines: A real-time line segment detector with a false detection control." *Pattern Recognition Letters* 32.13 (2011): 1633-1642.

[7] Kultaken, P., L. Xu, and E. Oja. "A new curve detection method: Randomized Hough transform." *Pattern Recognition Letters* 11 (1995): 331-338.

[8] Lowe, David G. "Distinctive image features from scale-invariant keypoints", International journal of computer vision 60.2 (2004): 91-110.

[9] Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. "Surf: Speeded up robust features." *Computer vision–ECCV 2006*. Springer Berlin Heidelberg, 2006. 404-417.

[10] J. Matas, O. Chum, M. Urban, and T. Pajdla. "Robust wide baseline stereo from maximally stable extremal regions." *Proc. of British Machine Vision Conference*, pages 384-396, 2002.

[11] Kadir, Timor, Andrew Zisserman, and Michael Brady. "An affine invariant salient region detector." *Computer Vision-ECCV 2004*. Springer Berlin Heidelberg, 2004. 228-241.

[12] Mikolajczyk, Krystian, et al. "A comparison of affine region detectors."*International journal of computer vision* 65.1-2 (2005): 43-72.

[13] Kang, Dong Joong, Jong Eun Ha, and In So Kweon. "Fast object recognition using dynamic programming from combination of salient line groups." *Pattern Recognition* 36.1 (2003): 79-90.

[14] Zhang, Lilian, and Reinhard Koch. "An efficient and robust line segment matching approach based on LBD descriptor and pairwise geometric consistency." *Journal of Visual Communication and Image Representation* 24.7 (2013): 794-805.

[15] Ferrari, Vittorio, Frederic Jurie, and Cordelia Schmid. "From images to shape models for object detection." *International journal of computer vision* 87.3 (2010): 284-303.

[16] Schindler, Konrad, and David Suter. "Object detection by global contour shape."*Pattern Recognition* 41.12 (2008): 3736-3748.

[17] Zhu, Qihui, et al. "Contour context selection for object detection: A set-to-set contour matching approach." *Computer Vision–ECCV 2008*. Springer Berlin Heidelberg, 2008. 774-787.

[18] Lindeberg, T. ‘Edge Detection and Ridge Detection with Automatic Scale Selection’, *International Journal of Computer Vision*, 1998, 30(2), 117-156.

[19] Duda, Richard O., and Peter E. Hart. "Use of the Hough transformation to detect lines and curves in pictures." *Communications of the ACM* 15.1 (1972): 11-15.

[20] Rosten, Edward, and Tom Drummond. "Machine learning for high-speed corner detection." *Computer Vision–ECCV 2006*. Springer Berlin Heidelberg, 2006. 430-443.

[21] Rublee, Ethan, et al. "ORB: an efficient alternative to SIFT or SURF." *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011.

[22] OpenCV Library, http://opencv.org/, accessed September 2015.

[23] Lindeberg, Tony. Scale-space theory in computer vision. Springer, 1993.

[24] Mikolajczyk, Krystian, and Cordelia Schmid. "A performance evaluation of local descriptors." *Pattern Analysis and Machine Intelligence, IEEE Transactions on*27.10 (2005): 1615-1630.

[25] Papari, Giuseppe, and Nicolai Petkov. "Edge and line oriented contour detection: State of the art." *Image and Vision Computing* 29.2 (2011): 79-103.

[26] Bao, Paul, Lei Zhang, and Xiaolin Wu. "Canny edge detection enhancement by scale multiplication." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27.9 (2005): 1485-1490.

[27] Leutenegger, Stefan, Margarita Chli, and Roland Y. Siegwart. "BRISK: Binary robust invariant scalable keypoints." *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011.

[28] Alahi, Alexandre, Raphael Ortiz, and Pierre Vandergheynst. "Freak: Fast retina keypoint." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. Ieee, 2012.

[29] Baillard, Caroline, et al. "Automatic line matching and 3D reconstruction of buildings from multiple views." *ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery*. Vol. 32. 1999.

[30] Kang, D. J., Ha, J. E., Kweon, I. S. 'Fast object recognition using dynamic programming from combination of salient line groups'. *Pattern Recognition*, 2003, 36, (1), pp. 79-90.

[31] Schindler, K., Suter, D.: 'Object detection by global contour shape'. *Pattern Recognition*, 2008, 41, (12), pp. 3736-3748.

[32] Yao, Z., Li, H.: 'Tracking a detected face with dynamic programming'. *Image and Vision Computing*, 2006, 24, (6), pp. 573-580.

[33] 'Average Face'. http://www.uniregensburg.de/Fakultaeten/phil_Fak_II/Psychologie/Psy_II/beautycheck/english/durchschnittsgesichter/durchschnittsgesichter.htm, accessed 27 July 2015.

[34] Viola, P., and Jones, M.: 'Rapid object detection using a boosted cascade of simple features'. Computer Vision and Pattern Recognition, 2001. CVPR 2001.

Proceedings of the 2001 IEEE Computer Society Conference on. Vol. 1. IEEE, 2001.

[35] Zhou, F., De la Torre, F. ‘Deformable graph matching’. *CVPR*, 2013, pp. 2922-2929.

[36] Cho, M., Alahari, K., Ponce, J. ‘Learning graphs to match’. *ICCV*, 2013, pp. 25-32.

[37] Marr, David, and Ellen Hildreth. "Theory of edge detection." *Proceedings of the Royal Society of London B: Biological Sciences* 207.1167 (1980): 187-217.

[38] Torre, Vincent, and Tomaso A. Poggio. "On edge detection." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 2 (1986): 147-163.

[39] Ziou, Djemel, and Salvatore Tabbone. "Edge detection techniques-an overview." *Pattern Recognition and Image Analysis C/C of Raspoznavaniye Obrazov I Analiz Izobrazhenii* 8 (1998): 537-559.

[40] Rafael C. Gonzalez, and Richard E. Woods, “Digital Image Processing”, 3rd ed. Addison-Wesley, Reading, MA, 2008.

[41] Hasegawa, J. I., H. Kubota, and J. I. Toriwaki. "Automated construction of image processing procedures by sample-figure presentation." *Proceedings of the eighth international conference on pattern recognition*. 1986.

[42] Rosenfeld, Avi. "A nonlinear edge detection technique." *Proceedings of the IEEE* 58.5 (1970): 814-816.

[43] Bao, Paul, Lei Zhang, and Xiaolin Wu. "Canny edge detection enhancement by scale multiplication." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27.9 (2005): 1485-1490.

[44] Grigorescu, Cosmin, Nicolai Petkov, and Michel Westenberg. "Contour detection based on nonclassical receptive field inhibition." *Image Processing, IEEE Transactions on* 12.7 (2003): 729-739.

[45] Martin, David R., Charless C. Fowlkes, and Jitendra Malik. "Learning to detect natural image boundaries using local brightness, color, and texture cues."*Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.5 (2004): 530-549.

[46] Sharon, Eitan, Achi Brandt, and Ronen Basri. "Fast multiscale image segmentation." *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*. Vol. 1. IEEE, 2000.

[47] Pham, Tuan Q., and Lucas J. Van Vliet. "Separable bilateral filtering for fast video preprocessing." *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*. IEEE, 2005.

[48] Crow, Franklin C. "Summed-area tables for texture mapping." *ACM SIGGRAPH computer graphics* 18.3 (1984): 207-212.

[49] Richard P. Feynman, Robert B. Leighton, and Matthew Sands, "The Feynman lectures on physics", Mainly mechanics, radiation, and heat. Vol. 1. Basic Books, pp. P. 19.1 – 19.3, 2011.

[50] Konstantinos G. Derpanis, "Integral image-based representations", Department of Computer Science and Engineering York University Paper, 2007, Vol. 1 No. 2 pp. 1-6.

[51] Viola, Paul, and Michael J. Jones. "Robust real-time face detection."*International journal of computer vision* 57.2 (2004): 137-154.

[52] Veksler, Olga. "Fast variable window for stereo correspondence using integral images." *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*. Vol. 1. IEEE, 2003.

[53] Victor Podlozhnyuk , "Image convolution with CUDA". NVIDIA Corporation white paper, 2097(3), 2007

[54] Dirk-Jan Kroon, "Numerical optimization of kernel based image derivatives". Short Paper University Twente, 2009.

[55] Ballard, Dana H. "Generalizing the Hough transform to detect arbitrary shapes."*Pattern recognition* 13.2 (1981): 111-122.

[56] Olson, Clark F., and Daniel P. Huttenlocher. "Automatic target recognition by matching oriented edge pixels." *Image Processing, IEEE Transactions on* 6.1 (1997): 103-113.

[57] Borgefors, Gunilla. "Hierarchical chamfer matching: A parametric edge matching algorithm." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 10.6 (1988): 849-865.

[58] Gao, Yongsheng, and Maylor KH Leung. "Face recognition using line edge map." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 24.6 (2002): 764-779.

[59] Zhu, Manli, and Aleix M. Martinez. "Subclass discriminant analysis." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 28.8 (2006): 1274-1286.

[60] Jevtic, A., and Bo Li. "Ant Algorithms for Adaptive Edge Detection." *Search Algorithms for En-gineering Optimization, ISBN* (2012): 978-953.

[61] D. A. Forsyth and J. Ponce, “Computer Vision: A Modern Approach,”1st Edition, Prentice Hall, August 2002.

[62] F. Devernay, “A Non-Maxima Suppression Method for Edge Detection with Sub-Pixel Accuracy,” INRIA Research Report 11/1995; 2724.

[63] Lindeberg, Tony. "Feature detection with automatic scale selection."*International journal of computer vision* 30.2 (1998): 79-116.

[64] Mikolajczyk, Krystian, and Cordelia Schmid. "Indexing based on scale invariant interest points." *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*. Vol. 1. IEEE, 2001.

[65] Grabner, Michael, Helmut Grabner, and Horst Bischof. "Fast approximated SIFT." *Computer Vision–ACCV 2006*. Springer Berlin Heidelberg, 2006. 918-927.

[66] Zheng, Zhiqiang, Han Wang, and Eam Khwang Teoh. "Analysis of gray level corner detection." *Pattern Recognition Letters* 20.2 (1999): 149-162.

[67] Cooper, James, Svetha Venkatesh, and Leslie Kitchen. "Early jump-out corner detectors." *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*. IEEE, 1991.

[68] Mokhtarian, Farzin, and Riku Suomela. "Robust image corner detection through curvature scale space." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 20.12 (1998): 1376-1381.

[69] Li, B., Jevtic, A. J., Söderström, U., Ur Réhman, S., and Li, H.: 'Fast Edge Detection by Center of Mass', *Proceedings of the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing 2013*, pp. 103-110.

[70] Bo Li, Shafiq Ur Réhman, Ulrik Söderström, "Independent Thresholds on Multi-scale Gradient Images", *Proceedings of the 1st IEEE/IIAE International Conference on Intelligent Systems and Image Processing 2013.*

[71] Bo Li, Ulrik Söderström, Shafiq Ur Réhman, Haibo Li, "Restricted Hysteresis Reduce Redundancy in Edge Detection", *in Journal of Signal and Information Processing*, August 2013.

[72] Gonzalez, R. C.: 'Digital Image Processing' (Pearson Education India, 2009).

[73] Li, B., Li, H., Soderstrom, U.: 'Scale-invariant Corner Keypoints'. *Image Processing, 2014 IEEE International Conference on*, pp. 5741-5745. IEEE, 2014.

[74] Ferrari, V., Tuytelaars, T., Van Gool, L.: 'Object detection by contour segment networks'. *In Computer Vision–ECCV 2006*, Springer Berlin Heidelberg, pp. 14-28

[75] Fischler, M. A., Bolles, R. C.: 'Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography'. *Communications of the ACM*, 1981, 24, (6), pp. 381-395.

[76] Chum, O., Matas, J.: 'Matching with PROSAC-progressive sampling concensus'. *Proc. of the CVPR*, 2005.

[77] The BioID Face Database'. https://www.bioid.com/About/BioID-Face-Database, accessed 27 July 2015.

Digital Media Lab

Zhengrong Yao, Initialization of Model Based Coding. 2003, 1652-6295:1.

Apostolos Georgakis, Textual Information Retrieval: An Approach Based on Language Modeling. 2004. 1652-6259:2.

Jiong Sun, Wireless Video: Complexity Adaptability and Predictability. 2004. 1652-6295:3.

Zhengrong Yao, Model Based Coding: Initialization, Parameter Extraction and Evaluation, 2005, 1652-6295:4.

Hung-Son Le, Face Recognition Using Hidden Markov Model. 2005. 1652-6295:5.

Jiong Sun, Football on Mobile Phones: Algorithms, Architectures and Quality. 2006. 1652-6295:6.

Ulrik Söderström, Very Low Bitrate Facial Video Coding Based on Principal Component Analysis. 2006. 1652-6295:7.

Johannes Karlsson, Real-Time Video over Wireless Ad-Hoc Networks. 2006. 1652-6295:8.

Hung-Son Le, Face Recognition: A Signal View Based HMM Approach. 2008. 1652-6295:9.

Shafiq ur Réhman, Vibrotactile Rendering of Dynamic Media Signals: Analysis and Algorithms. 2008. 1652-6295:10.

Ulrik Söderström, Very Low Bitrate Video Communication: A Principal Component Analysis Approach. 2008. 1652-6295:11.

Shafiq ur Réhman, Expressing Emotions Through Vibration for Perception and Control. 2010. 1652-6295:12.

Johannes Karlsson, Wireless Video Sensor Network and Its Applications in Digital Zoo. 2010. 1652-6295:13.

Jean-Paul Kouma, Wyner-Ziv Coding for Large-Scale Face Image Retrieval. 2011. 1652-6295:14.

Farid Abedan Kondori, Human Motion Analysis for Creating Immersive Experience. 2012. 1652-6295:15.

Shahrouz Yousefi, Enabling Media Technologies for Mobile Photo Browsing. 2012. 1652-6295:16.

Bo Li, Pushing Edge Detection to the Limit, 2013. 1652-6295: 17.

Karin Fahlquist, Creating New Experience for Zoo Visitors by Using Media Techniques, 2014. 1652-6295: 18.

Farid Abedan Kondori, Bring Your Body into Action: Body Gesture Detection, Tracking, and Analysis for Natural Interaction. 2014. 1652-6295:19.

Muhammad Sikandar Lal Khan, Enabling Physical Action in Computer Mediated Communication: an Embodied Interaction Approach. 2015. 1652-6295: 20.