Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Informatics

Redundant Edges and Parallel Algorithms in Binary Irregular Graph Pyramids

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der Technischen Wissenschaften

by

M. Tech. Majid Banaeyan
Registration Number 01428332

to the Faculty of Informatics

at the TU Wien

Advisor: O.Univ.Prof. Dipl.Ing. Dr.techn. Walter G. Kropatsch

The dissertation has been reviewed by:

Prof. Jos Roerdink

Asso. Prof. Phuc NGO

Vienna, 1st March, 2024

Majid Banaeyan

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Erklärung zur Verfassung der Arbeit

M. Tech. Majid Banaeyan

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 1. März 2024

Majid Banaeyan

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Acknowledgements

As I come to the end of my Ph.D. journey, I am filled with gratitude for all the support, encouragement, and guidance I have received along the way.

First and foremost, I would like to dedicate my thesis to my late father, who could not be here to witness this achievement. His unwavering love and support gave me the strength to pursue this dream, and I will always be grateful for the sacrifices he made to provide me with the opportunities to succeed.

I would also like to express my heartfelt thanks to my mother, who has been a constant source of inspiration and motivation throughout this journey. Her unwavering support and encouragement have helped me overcome the many challenges I faced during my Ph.D.

I am deeply grateful to my supervisor, Prof. Walter G. Kropatsch, for his invaluable guidance, encouragement, and support throughout my PhD. His expertise, insights, and feedback have been instrumental in shaping my research and helping me to grow as a scholar. Moreover, Prof. Kropatsch's sharp mind, accuracy, and innovative ideas have always pushed me to think outside the box and strive for excellence. I am fortunate to have had such a dedicated and inspiring supervisor, and I am grateful for all that he has done to help me achieve my academic goals.

I would also like to thank my siblings, my brother Mehdi, and my sisters Marziyeh and Minoo, for their support, encouragement, and love throughout my Ph.D. Their unwavering support and encouragement gave me the strength to persevere through difficult times.

I am deeply grateful to my wife Faezeh, for her love, patience, and unwavering support throughout this period. Her constant encouragement and belief in me have been an inspiration and a driving force in my success.

Finally, I would like to thank my colleagues, Dr. Darshan Batavia and Dr. Jiří Hladůvka, for their valuable insights, feedback, and encouragement throughout my Ph.D.

Thank you to all who have supported me on this journey. I am truly grateful for your kindness, generosity, and encouragement.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Abstract

In the modern digital era, an astonishing volume of data is being produced across a diverse range of fields, encompassing biomedical imaging, document processing, geosciences, remote sensing, video surveillance, social media, machine learning, and artificial intelligence. The efficient processing of such expansive data necessitates the development of effective data structures and parallel algorithms with minimal complexity. Improvements in hardware and the increase in massively available processing elements have rendered parallel processing a compelling solution to accelerate algorithm execution.

Irregular pyramids represent a powerful hierarchical structure that progressively reduces data size across levels, enabling rapid extraction of global information. Owing to their logarithmic height relative to the input size and parallel architecture, these structures offer considerable efficiency. In irregular pyramids, the fundamental data structure is a general graph embedded in image space, unconfined to array structures. However, graphs as versatile representative tools may harbor many redundant edges, thus accumulating memory.

Conventional approaches to constructing irregular pyramids involve selecting a set of edges for contraction to produce a smaller graph at a higher level, which is then simplified by removing unnecessary edges. Contrarily, this thesis introduces a method dubbed *Remove then Contract* (RtC), that predicts and eliminates redundant edges prior to pyramid construction under certain conditions. This approach not only reduces memory requirements for data storage but also simplifies the pyramid construction process. It has been demonstrated that the upper bound of redundant edges is half of all edges in the input graph. Moreover, by defining a set of independent edges in irregular pyramids, operations can be performed in a fully parallel manner. Given a sufficient number of available processing elements, the algorithms proposed herein exhibit parallel complexity, where the only limiting factors are memory requirements or the number of available independent processing elements. However, as the number of available processing elements increases, the efficiency of these algorithms becomes increasingly pronounced. The thesis applies the constructed irregular pyramid to perform fundamental binary image operations with reduced computational complexity. Two pivotal methods, Connected Component Labeling (CCL) and Distance Transform (DT), which necessitate both local and global information, are examined. A new pyramidal parallel connected component labeling is proposed that reduces the computational complexity of CCL from sequential linear to

parallel logarithmic complexity, $O(log\, n)$, where n is the diameter of the largest connected component (CC) in the 2D binary image. This method not only performs the CCL task but also provides topological information between CCs containing inclusions and multi-boundaries. The DT is also computed with parallel logarithmic complexity, $O(log\, n)$, where n is the maximum diameter of the largest foreground region in the 2D binary image. Furthermore, by defining DT over other data structures, such as combinatorial maps and n-dimensional generalized maps, novel DTs with higher resolution and n different distances are introduced. The efficiency of the proposed methods is demonstrated through simulation and comparison with state-of-the-art methods.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Kurzfassung

Im modernen digitalen Zeitalter wird ein erstaunliches Datenvolumen in einer Vielzahl von Bereichen produziert, einschließlich biomedizinischer Bildgebung, Dokumentenverarbeitung, Geowissenschaften, Fernerkundung, Videoüberwachung, sozialen Medien, maschinellem Lernen und künstlicher Intelligenz. Die effiziente Verarbeitung solch umfangreicher Daten erfordert die Entwicklung von effektiven Datenstrukturen und parallelen Algorithmen mit minimaler Komplexität. Verbesserungen der Hardware und die Erhöhung der verfügbaren Verarbeitungselemente haben die Parallelverarbeitung zu einer überzeugenden Lösung gemacht, um die Ausführung von Algorithmen zu beschleunigen.

Unregelmäßige Pyramiden stellen eine leistungsfähige hierarchische Struktur dar, die die Datengröße über Ebenen hinweg progressiv reduziert und so eine schnelle Extraktion globaler Informationen ermöglicht. Dank ihrer logarithmischen Höhe im Verhältnis zur Eingangsgröße und ihrer parallelen Architektur bieten diese Strukturen erhebliche Effizienz. Bei unregelmäßigen Pyramiden ist die grundlegende Datenstruktur ein allgemeiner Graph, der im Bildraum eingebettet ist und nicht auf Array-Strukturen beschränkt ist. Graphen als vielseitige Repräsentationswerkzeuge können jedoch viele redundante Kanten beherbergen und so Speicher ansammeln.

Konventionelle Ansätze zum Bau unregelmäßiger Pyramiden beinhalten die Auswahl einer Kantenmenge zur Kontraktion, um einen kleineren Graphen auf einer höheren Ebene zu erzeugen, der dann durch Entfernen unnötiger Kanten vereinfacht wird. Im Gegensatz dazu führt diese Dissertation eine Methode namens *Remove then Contract* (RtC) ein, die redundante Kanten vor dem Bau der Pyramide vorhersagt und eliminiert unter gewissen bedingungen. Dieser Ansatz reduziert nicht nur den Speicherbedarf für die Datenspeicherung, sondern vereinfacht auch den Pyramidenaufbauprozess. Es wurde gezeigt, dass die obere Grenze der redundanten Kanten die Hälfte aller Kanten im Eingabegraphen ist. Darüber hinaus können durch die Definition einer Reihe unabhängiger Kanten in unregelmäßigen Pyramiden Operationen in vollständig paralleler Weise durchgeführt werden. Bei ausreichender Anzahl verfügbarer Verarbeitungselemente weisen die hier vorgeschlagenen Algorithmen eine parallele Komplexität auf, wobei die einzigen begrenzenden Faktoren den Speicherbedarf oder die Anzahl der verfügbaren unabhängigen Verarbeitungselemente sind. Mit zunehmender Anzahl verfügbarer Verarbeitungselemente wird die Effizienz dieser Algorithmen jedoch immer deutlicher.

Die Dissertation wendet die gebaute unregelmäßige Pyramide an, um grundlegende Binärbildoperationen mit reduzierter Rechenkomplexität durchzuführen. Zwei zentrale Methoden, das Connected Component Labeling (CCL) und die Distanztransformation (DT), die sowohl lokale als auch globale Informationen benötigen, werden untersucht. Ein neues pyramidenparalleles Verbundkomponenten-Labeling wird vorgeschlagen, das die Rechenkomplexität des CCL von sequentiell-linear auf parallele logarithmische Komplexität, $O(log\, n)$, reduziert, wobei n der Durchmesser der größten Verbundkomponente (CC) im 2D-Binärbild ist. Diese Methode führt nicht nur die CCL-Aufgabe durch, sondern liefert auch topologische Informationen zwischen CCs, die Einschlüsse und Mehrfachgrenzen enthalten. Die DT wird ebenfalls mit paralleler logarithmischer Komplexität berechnet, $O(log\, n)$, wobei n der maximale Durchmesser der größten Vordergrundregion im 2D-Binärbild ist. Darüber hinaus werden durch die Definition der DT über anderen Datenstrukturen, wie kombinatorischen Karten und n-dimensionalen generalisierten Karten, neuartige DTs mit höherer Auflösung und n verschiedenen Abständen eingeführt. Die Effizienz der vorgeschlagenen Methoden wird durch Simulation und Vergleich mit modernsten Methoden demonstriert.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Contents

Abstract **vii**

Kurzfassung **ix**

Contents **xi**

1 Motivations and outline of the thesis **1**
1.1 Motivations 1
1.2 Outline 5

I First Part **9**

2 Introduction to Irregular Graph Pyramids **11**
2.1 Hierarchical Structures 11
2.2 Regular Pyramids 12
2.3 Irregular Pyramids 13
2.4 Combinatorial Pyramid 16

3 Redundant Edges in Binary Irregular Pyramid **19**
3.1 Redundant Edges in Irregular Pyramids 19
3.2 Total Order of the vertices 21
3.3 Redundant Edges in Binary Images 23
3.4 Remove then Contract method 24

4 Parallel Algorithms **29**
4.1 Parallel Algorithms 29
4.2 Independent Elements 30
4.3 Parallel Edge Removal 31
4.4 Parallel Edge Contraction 32

5 Fundamental Operations in Binary Images **33**
5.1 Operations in Binary Images 33
5.2 Connected Component Labeling 34

5.3 Distance Transform 35

6 Experiments and Results **37**
6.1 Initialization and Pre-processing in Proposed Algorithms 37
6.2 Redundant Edge Elimination 37
6.3 Connected Component Labeling 38
6.4 Distance Transform 38

7 Concluding Remarks **39**
7.1 Conclusion 39
7.2 Future Work 40

Bibliography **43**

Second Part **51**

Paper A
Removing Redundancies in Binary Images **53**

Paper B
Parallel $O(log(n))$ Computation of the Adjacency of Connected Components **67**

Paper C
Fast Distance Transforms in Graphs and in Gmaps **81**

Paper D
Fast Labeled Spanning Tree in Binary Irregular Graph Pyramids **93**

Paper E
Distance Transform in Parallel Logarithmic Complexity **105**

List of Figures **115**

Appendix
Curriculum Vitae and Publications **117**

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

CHAPTER 1

Motivations and outline of the thesis

Treasure the love you receive above all. It will survive long after your good health has vanished.

Omar Khayyam

1.1 Motivations

In the subsequent section, four distinct motivations for utilizing the irregular pyramid in this thesis will be elaborated.

1.1.1 Exponential Growth of Data

The digital age has ushered in an era of exponential data growth. Back in 1992, approximately 100 gigabytes of data were being produced by human beings every day [DK18]. This rate rapidly escalated, with the same volume being generated each hour by 1997, and each second by 2002. By 2018, an astonishing 50,000 gigabytes of data were being produced every single second [DK18], a figure that may be increased exponentially to an estimated 382,000 gigabytes per second in 2023[1].

Such a colossal amount of data brings with it significant challenges in terms of efficient management and information extraction [BR14]. These challenges necessitate the use of optimized data structures and parallel processing algorithms. These tools not only facilitate the handling of vast volumes of data [Hil16], allowing for efficient search and retrieval operations but also enable the transformation of this sea of information

---

[1] https://www.statista.com/statistics/871513/worldwide-data-created/

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

into actionable knowledge. As the growth of data continues, the importance of these technological assets becomes increasingly crucial, extending to sectors as diverse as healthcare, finance, and environmental science [MSC13]. Moreover, these vast quantities of data serve as a cornerstone for artificial intelligence and machine learning, further underlining the necessity of effective management and efficient extraction techniques in the digital era.

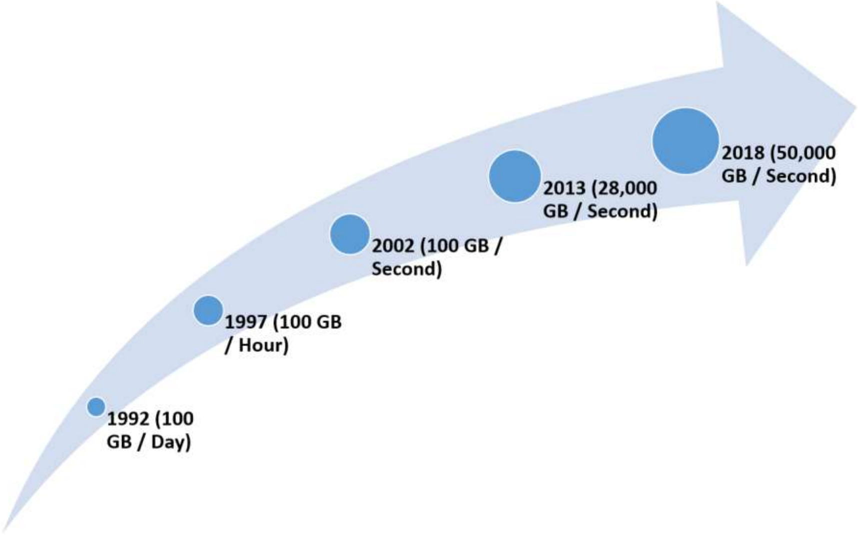


Figure 1.1: Growing Data volume and velocity [DK18].

1.1.2 Water's gateway to heaven

The Pattern Recognition and Image Processing Group (PRIP) initiated a collaborative project with two biology groups in 2020, entitled the *Water's Gateway to Heaven* project[2]. The objective of this project is to model the behavior of stomatal responses, specifically, to understand the opening and closing mechanisms of stomata in plant leaves. The modulation of stomatal aperture controls the influx of gases (e.g., CO_2) into leaves, a crucial aspect of the photosynthesis process.

The project involves the use of high-resolution 3D X-ray micro-tomography (μCT) and fluorescence microscopy. The dimensions of the 3D images exceed 2000 elements in each direction, leading to approximately $2000 \times 2000 \times 2000 \approx 2^{33}$ voxels per image. The vast volume of data, combined with the complexity of the models describing these processes, necessitates highly efficient data processing techniques. Additionally, since leaves are not entirely rigid but exhibit a degree of deformability, a **non-rigid structure** (like graphs) may be a suitable choice for modeling this problem.

---

[2] https://waters-gateway.boku.ac.at/

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

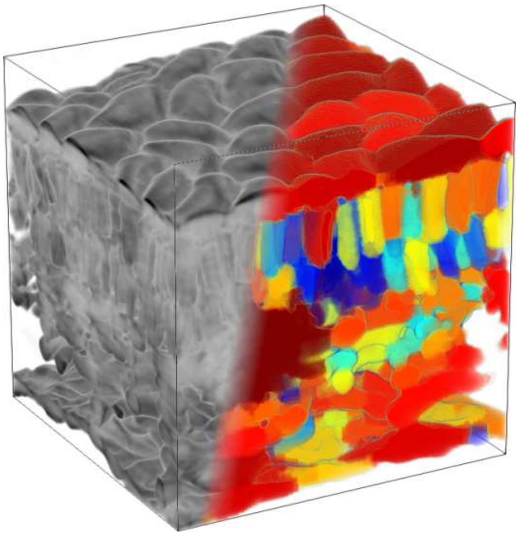


Figure 1.2: A small section of a 3D labeled image from the Water's Gateway to Heaven project [PVT+22].

1.1.3 Biological Perception

The task of perceptual recognition and description of complex real-world objects poses an immense challenge [FB82]. It is estimated that around half of the billions of neurons in the brains of higher mammals are dedicated to perception, a process that transpires at an astonishing speed [Uhr87].

Impressively, the perceptual system [Gra99] can recognize and describe complex scenes composed of intricate objects in a timeframe ranging from 30 to 800 milliseconds [Uhr87]. Considering that a single neuron's basic cycle time – the duration required to bridge the synapses over which neurons trigger other neurons – is approximately 1 to 2 milliseconds [Uhr86], the perceptual system operates with a serial depth of merely a few hundred steps at most, and potentially as few as a couple dozen. This signifies that our perceptual system operates predominantly as a **massively parallel** and shallowly serial system. This massive parallelism converges in logarithmic complexity towards the decision-making locus. To address the vision problem, Uhr [Uhr86] proposed the use of a pyramidal data structure. More recently, Pizlo has asserted that the **pyramid** serves as a general model for human problem-solving [Piz22], suggesting that massive parallel processing is necessary for the recognition of a complex scene in an instant [PWHM14].

1.1.4 Human's Vision Sensors are Irregular

Each human eye has approximately 10 million cones and 100 million rods, key types of photoreceptor cells integral to the human visual system [WBT+19]. Concentrated in the central area of the retina, known as the fovea, cones are equipped to discern color and

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

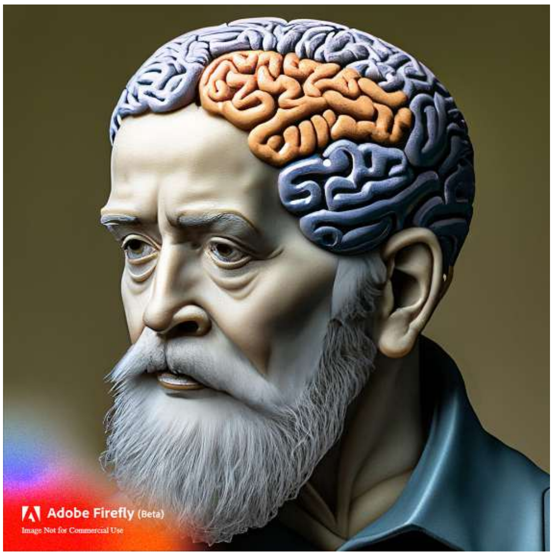


Figure 1.3: Human problem-solving. Image is generated by AI, https://firefly.adobe.com

shape [CSKH90]. The three types of cones each respond to different light wavelengths, enabling a vast color perception range. Conversely, rods are situated predominantly in the peripheral regions of the eye and are sensitive to intensity shifts spatially or temporally, making them more effective in low-light or nighttime conditions [CSKH90]. This distribution allows rods to serve as an initial alert system for changes in the visual field [Uhr14]. Both cones and rods play an essential role in transmitting visual stimuli from the retina, the light-responsive layer at the back of the eye, to the brain [Zek93].

While many neural network architectures assert biological plausibility, this claim generally only holds true in terms of basic signal processing functionality, such as weighted averages and activation functions [KB24]. However, this notion falls short when scrutinizing the underlying architecture of these networks. In particular, the sensors that provide visual input and the multitude of other sensors that deliver invaluable data to the human brain are **NOT** a **regular grid structure**. This observation starkly contrasts the prevalent artificial neural networks in current use, which typically employ regular grid-based architectures.

The arrangement of cones and rods, as shown in Fig. 1.4, is not array-like but rather characterized by non-uniform spatial sampling [Piz22]. The irregular distribution of these sensors is essential for human vision, as it optimally captures visual information across different regions of the retina [CSKH90]. This adaptive configuration allows the fovea, with its high cone density, to provide sharp and detailed central vision, while the abundance of rods in the peripheral retina enhances low-light sensitivity and peripheral awareness [WBT$^+$19].

This naturally irregular arrangement of sensors in the human eye necessitates the utiliza-

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

tion of data structures such as graphs. These structures are aptly suited to accurately represent the irregular embeddings inherent in biological vision systems. Moreover, they provide a platform to further investigate the potential advantages offered by these irregular sensor arrangements and how to emulate these complex structures in artificial systems to improve their performance and adaptability.

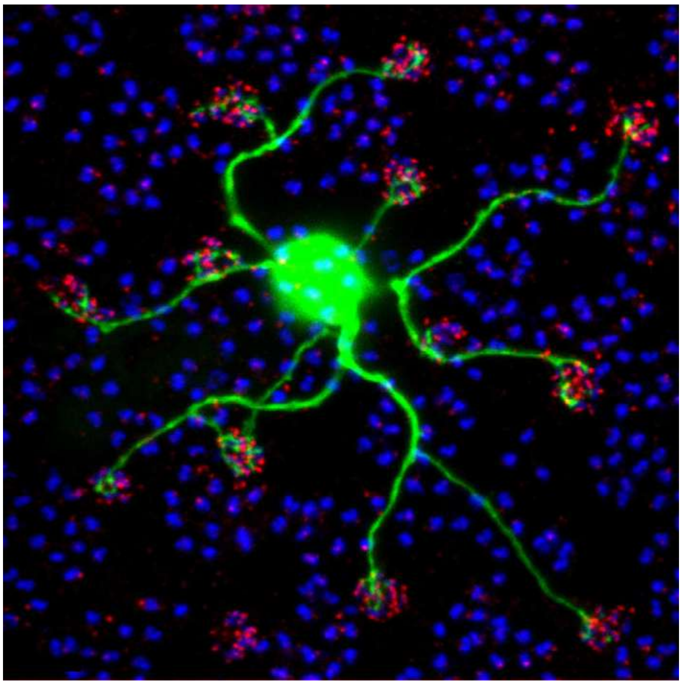


Figure 1.4: A segment of rods and cones in the human retina. Cones are represented by red dots, and Rods are depicted as blue dots. The green color illustrates bipolar cells that receive nerve signals from cones. Image courtesy of the National Eye Institute, National Institutes of Health (NEI/NIH).

1.2 Outline

This **CUMULATIVE** thesis, encompassing five selected research publications, focuses on the removal of redundancy and parallel algorithms in irregular pyramids. It is organized into three primary sections.

1.2.1 Part 1: Redundant Edges in Irregular Pyramids and Parallel Algorithms

In **Chapter 1**, the rationale for employing the irregular pyramid in this thesis is delineated through four main motivators. The following chapter, **Chapter 2**, provides an introductory exposition on hierarchical structures, particularly focusing on both regular

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

and irregular pyramids. This chapter elucidates the construction procedures of the irregular pyramids, detailing both the bottom-up and top-down movements.

Subsequently, **Chapter 3** presents the concept of redundant edges within the context of irregular pyramids, achieved through the introduction of total order over vertices. A new methodology for constructing the irregular pyramid is also briefly over-viewed in this section. **Chapter 4** transitions into parallel algorithms, focusing on their independent elements and expounding on the role of these parallel operations within the irregular pyramid framework.

Chapter 5 applies the principles of the irregular pyramid to present innovative approaches to compute fundamental operations in binary images, specifically in relation to connected component labeling (CCL) and distance transform (DT). Following this, **Chapter 6** provides a concise presentation of the conducted experiments and their subsequent results. The thesis concludes with **Chapter 7**, which encapsulates the final thoughts and potential avenues for future research.

1.2.2 Part 2: Selected Publications

Part 2 comprises **five papers** selected from my published works. These papers represent pivotal research contributions that introduce the concept of redundant structures in irregular pyramids and propose diverse parallel algorithms for implementing connected component labeling and distance transform operations using irregular pyramids. All five papers underwent peer review and have been accepted and published in the proceedings of international conferences (Papers A, B, and E), an international workshop (Paper C), or as an article in an international journal (Paper D).

1.2.2.1 Contributions

Over the course of my doctoral study, I authored **12 papers** and contributed to one **book chapter**, serving as the primary author for the papers and the secondary author for the book chapter. My major contribution was the development of a novel formalism that enables the prediction of **Redundant Edges** in **Binary Irregular Pyramids**.The summary of contributions for each paper is listed as follows:

Paper A:

- Introduces the concept of redundant edges in 2D binary irregular graph pyramids.
- Reduces the complexity of the edge removal operation in irregular pyramids from linear to parallel $O(1)$.
- Provides an upper bound for the number of redundant edges, proving that the number can be at most half of the total number of edges.
- Reduces the memory requirements for constructing $2D$ binary irregular pyramids.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Paper B:

- Introduces a novel method for constructing irregular graph pyramids, termed Remove then Contract (RtC).
- Proposes a new strategy for selecting contraction kernels in a connected plane graph by establishing a total order over its vertices.
- Introduces a parallel pyramidal connected component (//ACC) method, reducing parallel complexity to logarithmic in relation to the diameter of the largest connected component in the input binary image.

Paper C:

- Proposes a new methodology for computing the distance transform (DT) in connected plane graphs.
- Establishes a spanning forest of the foreground, allowing for distance propagation.
- Computes the geodesic distance transform (GDT) leveraging the hierarchical structure of irregular pyramids, reducing its complexity to parallel logarithmic complexity.
- Introduces the DT in n-dimensional generalized maps ($n - Gmaps$), defining n different distances within $n - Gmaps$ and elucidating their interrelations.

Paper D:

- Introduces the Fast Labeling Spanning Tree (FLST) method for computing the equivalent contraction kernels (ECK) of the input binary connected plane graph.
- Streamlines the selection of contraction kernels (CK) to only two steps and contracts them with parallel logarithmic complexity.

Paper E:

- Defines the DT within the combinatorial map structure.
- Computes the DT in $1D$ and $2D$ grid structures with parallel logarithmic complexity.

This achievement would not have been possible without the guidance and mentorship of Prof. **Walter G. Kropatsch**, my supervisor, and the head of the Pattern Recognition and Image Processing Group (PRIP). His assistance was instrumental in both the conception and the proper articulation of my idea. The valuable insights shared in our numerous discussions, which spanned hundreds of hours, substantially influenced the

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

contributions of my published works. Prof. Kropatsch’s critical scientific perspective and precise feedback have significantly shaped my academic trajectory.

In the PRIP lab, I also had the privilege of engaging in enriching discussions with Dr. Darsha Batavia and Dr. Jiří Hladůvka. These discussions culminated in our collaborative work on Papers A and C. Lastly, I am deeply indebted to my wife, Mrs. Faezeh Moteabbed. As a professional graphic designer, her assistance in creating the illustrations for my papers was invaluable.

1.2.3 Part 3: Appendix

The appendix includes the pseudo-code for the algorithms introduced in Papers A, B, C, and D. Additionally, it features my curriculum vitae and a comprehensive list of my publications pertinent to the field of irregular pyramids and their applications.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Part I

First Part

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

CHAPTER 2

Introduction to Irregular Graph Pyramids

> *In nature, we find patterns, designs and structures from the most minuscule particles, to expressions of life discernible by human eyes, to the greater cosmos.*
>
> ***Belsebuub***

The following section provides the fundamental definitions and concepts needed to comprehend the papers in Part 2. It discusses hierarchical structures in Section 2.1, followed by regular pyramids in Section 2.2. Section 2.3 offers more in-depth information about irregular pyramids, including details on bottom-up construction and top-down reconstruction, the extended region adjacency graph, and contraction kernel selection. Finally, Section 2.4 delves into the concept of combinatorial pyramids.

2.1 Hierarchical Structures

Visual data, distinguished by its vast quantity and high redundancy, necessitates the implementation of effective organizational and aggregation strategies [Hax07]. Such data typically exhibit spatial and temporal clustering of relevant information, emphasizing the need for systems that can manage both computational complexity and the conversion of raw data into symbolic descriptions. A hierarchical architecture [JR12], embodying a pyramid-like structure, caters to these demands, with the image forming the base and each successive layer representing increased abstraction levels.

Crucially, this architecture accommodates the dynamic flow and transformation of information across and between various layers. Two key mechanisms—bottom-up and top-down processing—facilitate this [Ros86]. *Bottom-up* or *fine to coarse* processing primarily involves the recursive transformation of localized data into a more encompassing, global format [Hax07]. This process aids in extracting salient features from an image and in data compression as the information ascends the hierarchy.

Contrastingly, *top-down* or *coarse to fine* processing involves a model-guided, non-uniform search of image data, aiming to validate or disprove the existence of specific structures. This procedure enhances the feature extraction conducted by bottom-up processes by disseminating abstract, high-level information to the lower levels of the hierarchy. In effect, the pyramid-like hierarchical architecture elegantly balances the requisites of parallel data processing and multi-resolution image representation [Duf86]. By integrating these distinct processes, the architecture can manage and interpret complex visual data in a more efficient manner.

2.2 Regular Pyramids

Introduced in the 1980s [BHR81, Ros83], regular image pyramids serve as a sequence of decreasing resolution images, beneficial for image processing and analysis due to their computational efficiency and their capacity for simultaneously processing both local and global features [Kro87]. Their systematic construction facilitates fast top-down access to every pixel of the original image and enables the entire pyramid to be built efficiently [BCR90]. However, their regular structure induces drawbacks [Kro02], such as the limited encoding of regions at a given level and significant alterations due to minor initial image shifts [BK12].

In this structure, each level exhibits exponentially decreasing resolution, from the base representing the original image to the top, usually a single pixel averaging base pixels. The term **reduction window** refers to the relationship connecting any pyramid pixel to a set of pixels at the level below [Mee89a]. Conversely, the **receptive field** refers to the set of base-level pixels associated with a given pyramid pixel [MMR91]. A constant **reduction factor** is maintained between any two successive pyramid levels, defining the size ratio between these images.

Despite beneficial properties like noise robustness, the transformation of global base image properties into local features at higher levels, and the facilitation of object detection, regular pyramids have limitations [JM92]. These include sensitivity to minor image shifts (the shift dependence problem), artificial bounding of region encoding due to fixed reduction window sizes, and a failure to preserve the image's topological structure [SMK10]. To mitigate these issues, irregular pyramids were introduced as an alternative, which will be detailed in the following section.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

2.3 Irregular Pyramids

Irregular pyramids, as first proposed by Meer [Mee89a] and Montanvert et al. [MMR91], sought to retain the significant advantages of regular pyramids while addressing their key limitations. Regular pyramids possess two advantageous properties over non-hierarchical image processing algorithms: their ability for bottom-up parallel computation, where each pixel determines its value independently from its child, and the pyramid height's relationship with the logarithm of the image size due to a fixed decimation ratio. These properties enable any pyramid to be computed in $O(\log |n|)$ parallel steps [HGS$^+$02], where $|n|$ is the number of pixels in the image. Despite these merits, the rigidity of regular pyramids, evident in the fixed shape of reduction windows and static neighborhood of each pixel, impedes the pyramid's adaptability to data and preservation of adjacency relationships across the pyramid. To circumvent these issues, irregular pyramids are formulated as progressively reduced graph stacks.

Irregular pyramids consist of a series of progressively smaller graphs [Kro95]. Each graph in this series is termed a ***level*** of the pyramid, with the lowest level known as the base graph, which aligns with the input image. In this base graph, pixels correspond to the vertices of the graph, and two vertices are connected by an edge if the respective pixels are 4-connected, meaning they are adjacent either horizontally or vertically. This concept gives rise to what is referred to as the neighborhood graph, $G(V, E)$, of the image. Opting for a 4-neighborhood model, where each pixel is connected to its four immediate neighbors (left, right, top, bottom), is preferred as it avoids intersecting edges between diagonally placed pixels in a 2×2 matrix. This maintains the planarity of the graph, in contrast to an 8-connected model, where pixels are also connected to their diagonal neighbors, potentially leading to intersecting edges and disrupting graph planarity[Kle14].

Definition 1 (Plane Graph)**.** *A plane graph is a graph embedded in the plane such that its edges intersect only at their endpoints [Tru93].*

In the plane graph there are connected spaces between edges and vertices and every such connected area of the plane is called a ***face***. The ***degree*** of a face is quantified by the number of edges that form its boundary. A face that is enclosed by a cycle is designated an ***empty face***. Conversely, a *non-empty* face implies that traversing the boundary would require visiting vertices or edges twice [Kle14]. An empty face that includes only one edge is termed an ***empty self-loop***. Consider an empty face with a degree of 2, which consists of two edges sharing the same endpoints. These parallel edges are designated as ***multiple edges***. Essentially, multiple edges are those that connect the same endpoints, illustrated as edges $e_{u_1,v_1} \neq e_{u_2,v_2} \neq e_{u_3,v_3}$, where $u_1 = u_2 = u_3$ and $v_1 = v_2 = v_3$.

A graph G that includes parallel edges and/or self-loops is defined as a ***multiple graph***, whereas a graph without parallel edges or self-loops is termed a ***simple graph***. Vertices at the ends of the same edge are considered *neighbors*, and edges sharing a common vertex are ***adjacent***. A non-empty graph is ***connected*** when there exists a path linking

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

any two of its vertices. The ***spanning tree*** of a graph is a tree that encompasses all the graph's vertices.

The development of irregular pyramids relies on two critical graph operations: ***edge contraction*** and ***edge removal***. ***Edge contraction*** merges two adjacent vertices into one, reassigning all related edges to the new vertex, thereby preserving connectivity [Kro95, Kro98]. Conversely, ***edge removal*** simply eliminates an edge without affecting the graph's vertices or the remaining edges' connections (refer to Fig. 2.2). Within this framework, vertices or edges absent in subsequent pyramid levels are labeled as ***non-surviving***, whereas those that continue to the next level are ***surviving***. The distinction between surviving and non-surviving elements is made through the identification of a contraction kernel.

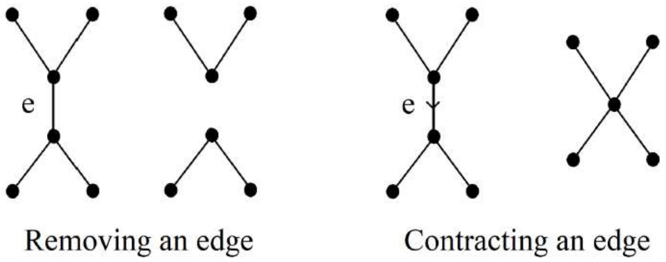


Figure 2.1: Edge Operations [BK22b]

Definition 2 (Contraction Kernel [KB24]). *Suppose we have an input graph $G(V, E)$ to be contracted. A contraction kernel $K \subset E$ is a subset of edges forming a spanning forest within G. Each constituent tree of this forest encompasses one surviving vertex, and in certain extreme cases, the tree could manifest as a single (surviving) vertex.*

2.3.1 Bottom-up Construction

The initiation of an irregular pyramid construction requires a 4-neighborhood connected plane graph, represented as $G = (V, E)$. Transitioning to a smaller graph at a higher level involves selecting a set of contraction kernels—specific edges whose contraction reduces the graph's size by decreasing both vertices and edges. This process may introduce multiple edges or self-loops, necessitating the removal of some edges to streamline the resulting graph. The characteristics of the vertices and edges in the condensed graph are determined through ***reduction functions***. These procedural steps are iteratively applied to build successive pyramid levels until a predetermined termination criterion is achieved, marking the pyramid's completion. With a reduction factor of at least 2, the pyramid's height is ultimately limited by the logarithm of the base graph's diameter.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Wien Bibliothek Your knowledge hub

2.3.2 Top-down Reconstruction

Top-down reconstruction process begins from the apex of the pyramid, where we have an abstract encapsulation of the image's visual features along with their spatial and topological aspects containing inclusion relationship and multi-boundaries. To unlock the insights derived from the input image, we cascade downwards through the pyramid's levels, reaching the base image. This downward journey helps illuminate the features and entities unearthed at higher levels, facilitated by reversing the construction steps from the bottom-up process. This is made possible through a ***canonical*** model proposed by [TK14a], which meticulously records contraction kernels and simplification parameters in chronological order. This model also includes mechanisms to undo edge contractions and removals, while preserving input attributes for the reduction function.

2.3.3 Extended Region Adjacency Graph

Regions in an image are defined as maximally connected components, consisting of adjacent pixels that either share the same group, determined by a defined similarity metric, or in the case of labeled images, have identical labels. Image segmentation assigns a label to each pixel, identifying sets of pixels sharing similar properties. The adjacency of these regions is typically represented by the *region adjacency graph (RAG)*. In RAG, each vertex represents a connected set of pixels with identical labels, and two vertices connect if their corresponding regions, with different labels, share a common boundary.

However, the standard RAG, typically viewed as a simple graph devoid of multiple edges and self-loops, fails to accurately describe all possible topological configurations between regions. To address this shortcoming, Kropatsch [KB24] introduced an **extended** version, the **E-RAG**.

Definition 3 (Extended Region Adjacency Graph (E-RAG))**.** *An E-RAG is a Region Adjacency Graph that includes multiple edges and self-loops.*

The E-RAG provides a refined depiction of topological relations, allowing for the representation of regions contained within or enclosing others, or instances where multiple disjoint boundaries exist between any two regions. Furthermore, it can explicitly model the region outside the image, ensuring every region on the image border is adjacent to it.

2.3.4 Selecting the Contraction Kernels

Selecting the contraction kernels is crucial for the construction of the irregular pyramid. For efficient parallel contraction, these kernels must be independent of one another. This independence facilitates the use of massively parallel processing, significantly enhancing the construction of the irregular pyramid. Ideally, contraction kernels should be small trees with a depth of one for optimal efficiency. Various methods for selecting contraction kernels include the Maximal Independent Vertex Set (MIS) proposed by Meer [Mee89b],

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

the Maximal Independent Edge Sets (MIES), and the Maximal Independent Directed Edge Sets (MIDES) introduced by Kropatsch [KHPL05].

In MIS, a collection of maximally independent vertices is formulated through an iterative stochastic mechanism. This process entails assigning to each vertex a random variable, uniformly distributed across a range from 1 to n, where n represents the total number of vertices within the input-connected plane graph. Each vertex thereby receives a unique number. Within this framework, vertices that attain a local maximum of this assigned variable are designated as survivor vertices, distinguishing them from their neighboring vertices, which are categorized as non-survivors [Mee89b]. Extending this concept, MIES applies the principles of MIS to an edge-graph [Chr75] derived from the original graph G [KHPL05]. This edge-graph is intricately constructed from G, with each vertex within the edge graph mirroring an edge in G, and connectivity between two vertices in the edge-graph is established based on the incidence of their corresponding edges in G to the same vertex. In MIES, edges undergo contraction in both directions. Conversely, the MIEDS method is tailored for cases necessitating directional constraints on edge contraction. In MIEDS dealing with edges instead of vertices, the contraction kernels may be selected in the same manner as in MIS [KHPL05].

2.4 Combinatorial Pyramid

The combinatorial pyramid [BK01] is a hierarchy of successively reduced combinatorial maps (CMs). A combinatorial map, resembling a graph, stores explicitly the orientation of edges around each vertex. It is defined by a triple $G = (D, \alpha, \sigma)$, where D is a finite set of darts [BK12]. A dart considered a half edge, forms the foundational element in the CM structure. The involution α on the set D forms a one-to-one mapping between consecutive darts on the same edge, i.e., $\alpha(\alpha(d)) = d$. The permutation σ on the set D encodes consecutive darts around a vertex, turning counterclockwise [TK14b]. Note that the clockwise orientation is denoted by σ^{-1}. Fig. 2.2a illustrates a set of adjacent darts with their respective σ and α encodings. Here, the edge e between two vertices u and v is represented as $e = (d, \alpha(d))$, where $u, v \in V$ and $e \in E$. V and E denote the sets of vertices and edges of the graph $G = (V, E)$, respectively.

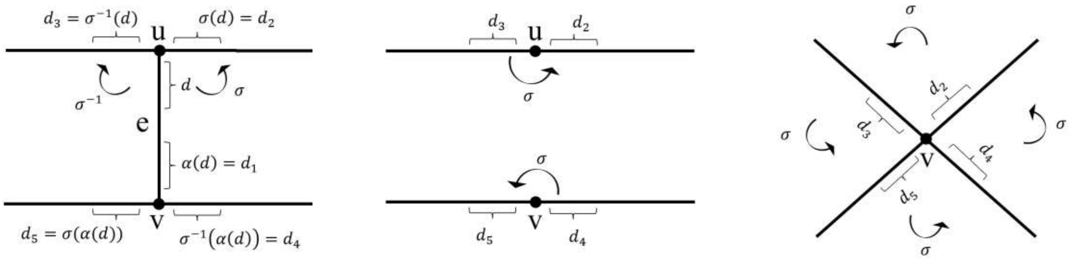
The removal and contraction operations in a combinatorial pyramid are defined as follows:

Definition 4 (Removal Operation)**.** *The removal operation eliminates one edge, denoted as $G\backslash e$, and modifies the adjacent darts as:*

$$\sigma(\sigma^{-1}(d)) = \sigma(d), \quad \sigma(\sigma^{-1}(\alpha(d))) = \sigma(\alpha(d)) \tag{2.1}$$

Definition 5 (Contraction Operation)**.** *The contraction operation eliminates one edge, represented as G/e, collapses its two endpoints, and modifies the adjacent darts as:*

$$\sigma(\sigma^{-1}(d)) = \sigma(\alpha(d)), \quad \sigma(\sigma^{-1}(\alpha(d))) = \sigma(d) \tag{2.2}$$



(a) (b) (c)

Figure 2.2: Two main operations in irregular graph pyramids. (a) Before applying an operation. (b), (c) after applying the operations [**?**]. (a) An edge e with its incident darts in the CM. (b) Removal operation, $G \backslash \{e\}$. (c) Contraction operation, $G/\{e\}$.

Fig.2.2b and Fig.2.2c depict the removal and contraction operations in the combinatorial map. Importantly, the contraction operation maintains the graph's **connectivity** [Kro95].

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

CHAPTER 3

Redundant Edges in Binary Irregular Pyramid

> *Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away.*
>
> ***Antoine de Saint-Exupery***

*This chapter delves into the concept of redundant edges in irregular pyramids (**Paper A, D**). Section 3.1 outlines the definition of redundant edges. The idea of the total order of vertices is explored comprehensively in Section 3.2. Section 3.3 characterizes redundant edges within binary irregular pyramids. Section 3.4 provides a comparison of a novel method for constructing irregular pyramids with prevalent prior methods. The specifics of the new approach are elaborated upon in **Paper B**.*

3.1 Redundant Edges in Irregular Pyramids

Redundant information within a structure presents two primary challenges. Firstly, it can consume extra memory space, leading to inefficient use of storage. Secondly, it can instigate unnecessary processing, thereby reducing the overall efficiency of associated algorithms. In the case of irregular pyramids, these redundancies are embodied by the multiple-edges and empty self-loops. These superfluous structures are generated post-contraction at each level of the pyramid.

Imagine an empty face with a degree of n, containing n vertices and n edges (see Fig. 3.1a). The contraction of one edge will result in a face with a degree of $n-1$. By iteratively contracting edges, the diminishing face ultimately becomes a triangle with a degree of 3. Contracting an edge from this triangle results in a face with a degree of 2, forming a double edge. Contracting the double edge subsequently gives rise to a self-loop with a degree of 1. Fig. 3.1b illustrates two adjacent triangles that share one edge, where the contraction of one edge from each triangle generates multiple edges. Contraction of multiple edges creates self-loops.

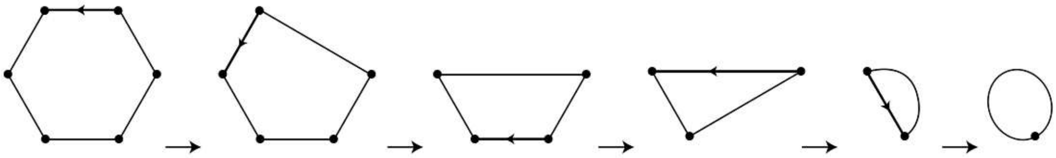
In construction of the irregular pyramids, multiple edges and self-loops are not topology relevant unless they enclose additional vertices or edges. To determine this, one can examine the dual graph $\overline{G}(\overline{V}, \overline{E})$, where each face of the original graph is represented as a vertex in $\overline{G}$, and edges in $\overline{G}$ correspond to shared edges between adjacent faces in the original graph. The degree of a face, represented as a vertex in the dual graph $\overline{V}$ facilitates the decision of the topological relevance of multiple edges and self-loops [KB24]:

Definition 6 (Topology-relevant)**.** *A face of the dual graph* $\overline{G}$ *is considered topology-relevant for* G *if its degree exceeds 2:* $deg(\overline{v}) > 2$ *for* $\overline{v} \in \overline{V}$*.*

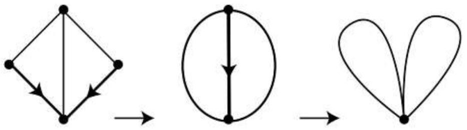
Definition 7 (Topology-irrelevant)**.** *A face that is not topology-relevant is topology-irrelevant.*

Definition 8 (Redundant edges)**.** *In a hierarchical structure, multiple edges and self-loops surrounding topology-irrelevant faces are redundant edges.*

Such redundant edges, being nonessential to topology, are not required to completely reconstruct the irregular pyramid. Self-loops can be eliminated without disconnecting any hole or substructure in the dual graph. Similarly, multiple edges can be removed as long as the final remaining edge is kept intact to preserve connectivity.



(a) Step-by-step contraction of an empty face with a degree of 6 until it transforms into an empty self-loop.



(b) Step-by-step contraction of two adjacent triangles.

Figure 3.1: Generation of multiple edges and self-loops.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

3.2 Total Order of the vertices

Traditional techniques for building irregular pyramids [Kro95, BK00, HIK06, CJGK16] aim to eliminate redundant edges that form *after* contractions during the simplification stage. These irregular pyramids use various methods such as MIS [Mee89b], MIES [KHPL05], and MIEDS [HGK03] to choose the contraction kernels (CKs). However, these edge selection techniques choose CKs in a *random* manner. This situation raises an important question: How can we exert more *control* over the CK selection process? Moreover, is it feasible to anticipate redundant edges before constructing the irregular pyramid? The concept of establishing a *total order* for the vertices of the neighborhood graph addresses these questions.

Definition 9 (Equivalent Contraction Kernel (ECK))**.** *An Equivalent Contraction Kernel (ECK) refers to a spanning tree that combines all contraction kernels into a single contraction kernel which generates the same result in one single contraction.*

The detailed definition of ECK can be found in [Kro98, HSGK01].

Definition 10 (Root of ECK)**.** *The root of ECK of the receptive field is its surviving vertex at a higher level of the pyramid.*

Definition 11 (Total Order Function)**.** *A Total Order (TO) function is a* ***bijective function*** *that assigns a unique number from* 1 *to* n *to each vertex of a connected plane graph composed of* n *vertices.*

$$\begin{aligned} TO : \{v_1, v_2, v_3, ..., v_n\} &\rightarrow \{1, 2, 3, ..., n\} \\ TO(v_i) = j \quad & i, j \in \{1, 2, 3, ..., n\} \\ TO(v_i) = TO(v_k) &\iff i = k \end{aligned} \tag{3.1}$$

With a Total Order [Hal60] in hand, an *equivalent contraction kernel function* (ECKF) is defined over the vertices of the receptive field. This function ensures that combining all contraction kernels forms the spanning tree of the receptive field.

Definition 12 (Equivalent Contraction Kernel Function (ECKF))**.** *An ECKF is a* ***surjective function*** *applied over a Total Order. It associates every non-surviving vertex with its respective surviving vertex within an incidence relationship such that combining all the CKs forms the spanning tree of the receptive field.*

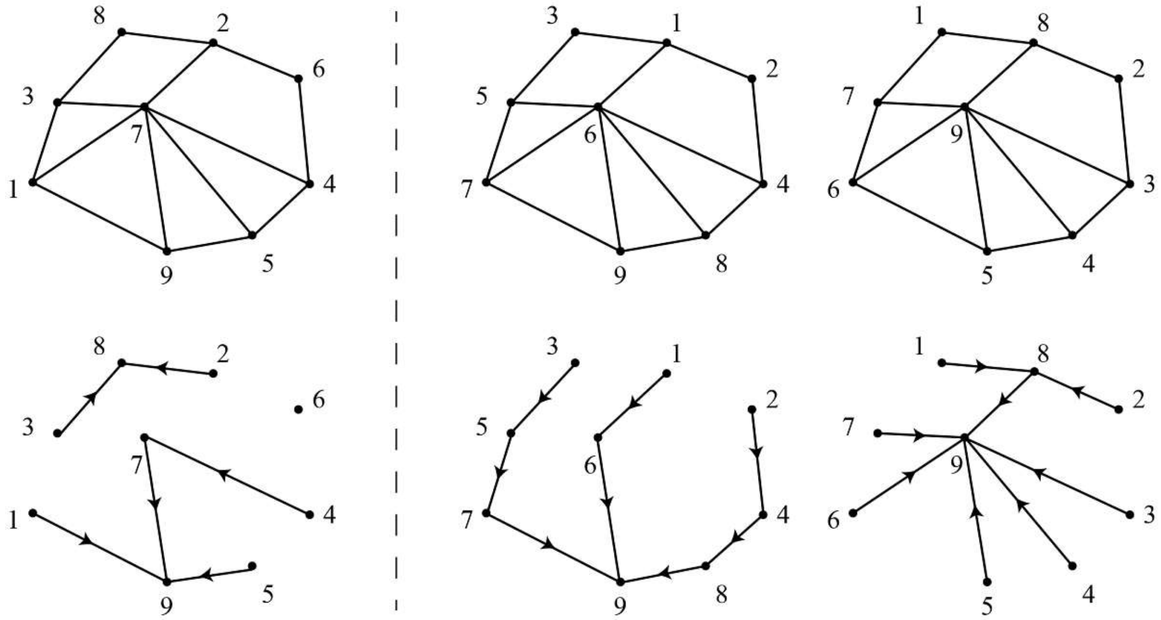
$$\begin{aligned} ECKF : \{v_1, v_2, v_3, ..., v_n\} &\rightarrow \{v_1, v_2, v_3, ..., v_n\} \\ \forall v_i \in \{v_1, v_2, v_3, ..., v_n\}, \exists v_j \in \{v_1, v_2, v_3, ..., v_n\},\ & ECKF(v_i) = v_j \end{aligned} \tag{3.2}$$

A function that assigns the **maximum** value among its neighboring vertices to each vertex could be considered a suitable candidate for the ECKF. However, this requires defining a proper, *valid* total order in advance.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Definition 13 (Valid Total Order)**.** *A Valid Total Order is a total order that, upon applying the Equivalent Contraction Kernel Function (ECKF), yields a spanning tree of the receptive field.*

For instance, applying the maximum function directly to the vertices in the image shown in Fig. 3.2 does not qualify as an ECKF due to the lack of a valid total order. Conversely, when valid total orders are assigned, the maximum function effectively serves as the ECKF.



(a) a non-valid total order

(b) valid total orders

Figure 3.2: A non-invalid total order (left) fails to generate a spanning tree, while valid total orders (right) successfully yield spanning trees.

With a given set of n vertices, there can be numerous total orders. However, when selecting a function to serve as the ECKF, our interest primarily lies in identifying valid total orders. This raises the question: how do we generate a valid total order? In this thesis, the *max* function is considered to be the ECKF, applied to the neighborhood of a vertex. The query now becomes: what are the valid total orders corresponding to this max function?

Proposition 1. *When the max function is chosen as a candidate for ECKF, a total order qualifies as a* ***valid total order*** *if and only if it possesses only one global maximum, contains no other local maxima, and assigns the maximum number, n, to the root.*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Proof. Consider a connected plane graph, $G = (V, E)$, with n vertices. Given the properties of total order, each vertex is assigned a unique number. This results in each vertex identifying only one neighboring vertex with a higher number. With the presence of only one global maximum and no other local maxima, all vertices, except for the root, can locate a different vertex from themselves by applying the ECKF. This ensures that after the ECKF application, no vertex remains isolated and the graph stays connected. Moreover, the uniqueness of the numbers assigned to vertices implies that a vertex selects only one neighbor, which eliminates the possibility of loops in the resulting connected graph. Hence, the resulting connected graph forms a spanning tree of the original graph, confirming the assigned total order as a valid total order. ☐

3.3 Redundant Edges in Binary Images

Paper A categorizes the graph's edges (E) into zero-edges (E_0) and one-edges (E_1) based on their contrast value $contrast(e) = |g(u) - g(v)|$, where $e = (u, v)$ and $g(v)$ represents the binary attribute (0 or 1) of each vertex $v \in V$ in G. This classification helps identify redundant edges as either redundant zero-edges (RE_0) or redundant one-edges (RE_1), depending on their placement in a face. Paper D extends this concept in connected component labeling, designating intra-CC edges within connected components as zero-edges, and inter-CCs edges between components as one-edges.

Paper A provides theoretical proof that demonstrates the **upper bound** of the number of redundant edges equal to up to **half** of the total number of edges of the grid at the base level. In the worst case, a binary neighborhood graph may not have any redundant edges. For instance, consider a binary image comprising a checkerboard pattern (Fig. 3.3a). The corresponding neighborhood graph (Fig. 3.3b) only consists of inter-CCs edges. Consequently, according to the definitions of redundant edges, there are no redundant edges in this case.

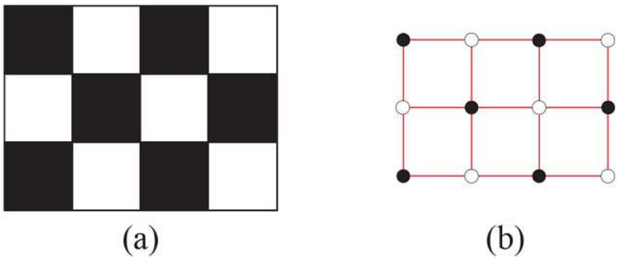

(a) (b)

Figure 3.3: An example of a binary image showcasing a checkerboard pattern, where its corresponding neighborhood graph contains no redundant edges.

Alg. 1 details the process of removing redundant edges, as discussed in Papers A and B. Step 3 of the algorithm partitions the edges of the graph (E) into zero-edges (E_0) and one-edges (E_1). This step is crucial for identifying redundant edges in Steps 5 and 6,

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

where each edge examines its corresponding face of degree 4 to determine redundancy. Every step of the algorithm (Steps 3 to 8) operates with parallel constant complexity, thus granting the entire algorithm parallel constant complexity.

Algorithm 1 Removing Redundant Edges

```
Input: A 4-connected neighborhood graph $G = (V, E)$ of a binary input image
Initialization: Define the total vertex order (equations (1) and (2) in Paper A)
$E = E_0 \dot{\cup} E_1$
Select contraction kernels from $E_0$ (equation (4) in Paper A)
Identify redundant zero-edges ($RE_0$)
Identify redundant one-edges ($RE_1$)
Determine independent redundant edges
Remove independent redundant edges to simplify the graph
Output: A simplified connected plane graph $G' = (V', E')$
```

3.4 Remove then Contract method

The development of the **Remove then Contract** (RtC) method marks a significant advancement in constructing irregular pyramids by identifying and eliminating redundant edges with parallel constant complexity, in contrast to traditional methods where edge removal complexity follows the inverse of the *Ackermann* function [Wil95]. Unlike previous methods where edge removal was more complex than contraction operations, RtC achieves both tasks with improved efficiency: edge removal is executed with constant complexity, and contraction operations maintain **parallel logarithmic complexity**, $O(\log n)$, where n is the input graph's diameter. This innovation streamlines the construction of binary irregular pyramids, making the process more efficient.

Alg. 2 outlines the RtC method. In Step 4, a spanning forest of contraction kernels is created, with each tree within the forest being independent of the others. Proposition 2 in Paper B proves that contracting these kernels (Steps 7 and 8 within the while loop) exhibits parallel logarithmic complexity, denoted as $log\delta(CK)$, where $\delta(CK)$ represents the diameter of the largest connected component's contraction kernel in the image.

Fig. 3.4 presents an example of constructing the binary irregular pyramid using traditional methods [Kro95, BK00, CJGK16]. In these methods, the contraction kernels are incrementally selected at each pyramid level. The edges of the contraction kernels are oriented edges, and the surviving vertices are depicted with a green circle around each of them. However, as a result of the edge contractions, the resulting smaller graph at higher levels contains redundant edges, necessitating simplification. The left parts of Fig. 3.4b,c,d,e depict the graphs before simplification, while the right parts show the graphs after simplification. Through repetitive contractions and simplification steps, the pyramid eventually reaches its apex, where E-RAG represents each connected component

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Algorithm 2 Remove then Contract (RtC)

```
1:  Input: A 4-connected neighborhood graph $G = (V, E)$ of a binary input image
2:  Initialization: Define the total vertex order (equations (1) and (2) in Paper A)
3:  $E = E_0 \dot{\cup} E_1$
4:  Select contraction kernels from $E_0$ (equation (4) in Paper A)
5:  while a contraction kernel exists do
6:      Remove redundant edges (using Algorithm 1, Steps 5 to 8)
7:      Identify a set of independent contraction kernels selected in Step 4
8:      Contract the identified independent contraction kernels.
9:  end while
10: Output: A simplified connected plane graph $G' = (V', E')$
```

by a vertex and the relationship between the connected components by edges. In Fig. 3.4e, the black edges encode the inclusion relationships between regions.

Fig. 3.5 demonstrates the construction of the binary irregular pyramid using the RtC method, wherein most of the contraction kernels are selected at the base level. In Fig. 3.5b, the left part showcases the detection of redundant edges before the contraction of the contraction kernels. These redundant edges are indicated by dotted lines. Upon removal of these redundant edges, the graph is simplified, as shown in the right part of Fig. 3.5b at the middle level. At this stage, the only remaining contraction kernel, denoted as **c**, is selected, and its contraction leads the pyramid to its apex. In Fig. 3.5c at the top level, edges **a** and **b** represent the inclusion relationships. It is worth noting that the topology of E-RAG in Fig. 3.4 and Fig. 3.5 remains the same, with only the surviving vertices at the top level differing.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

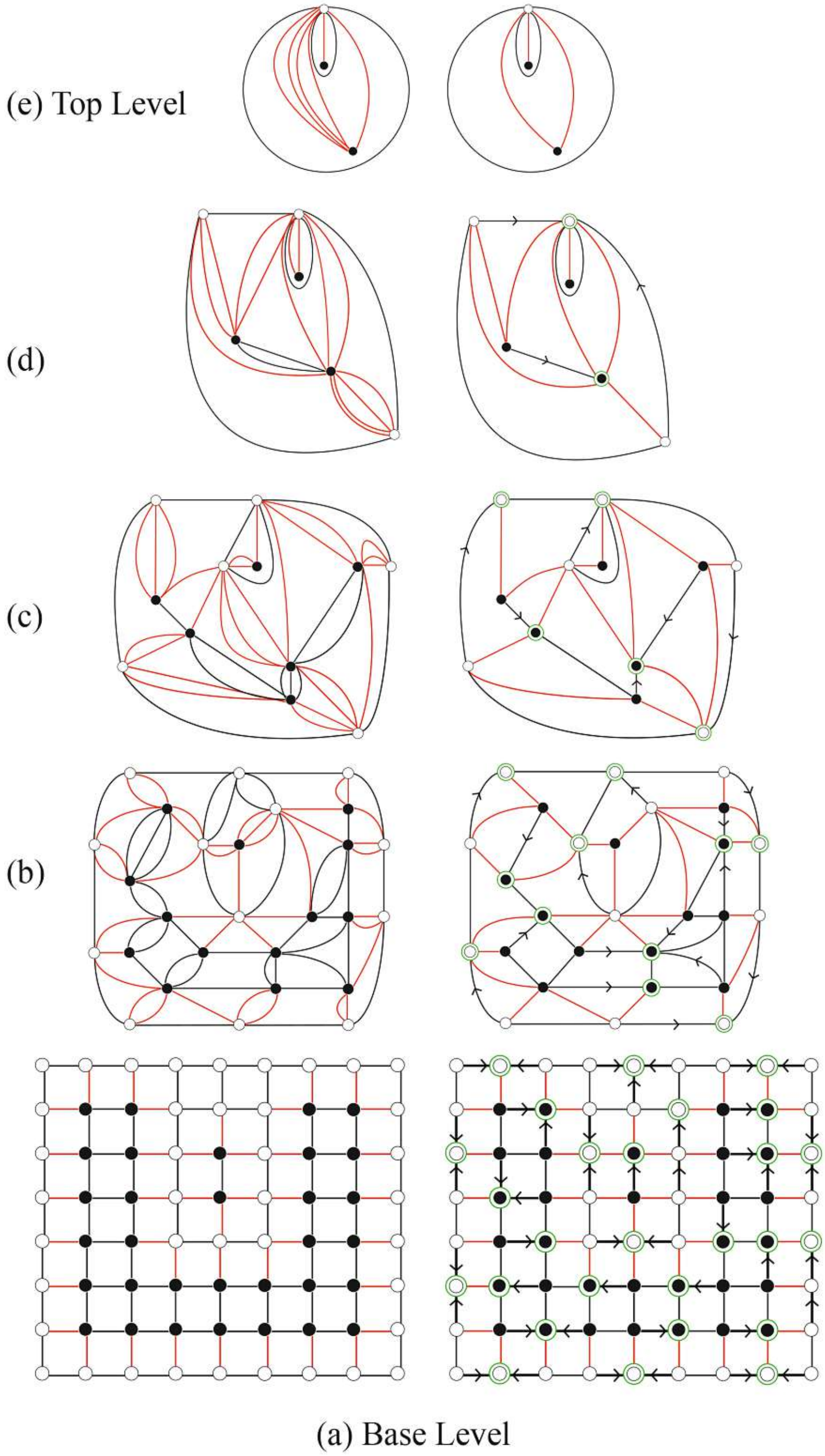
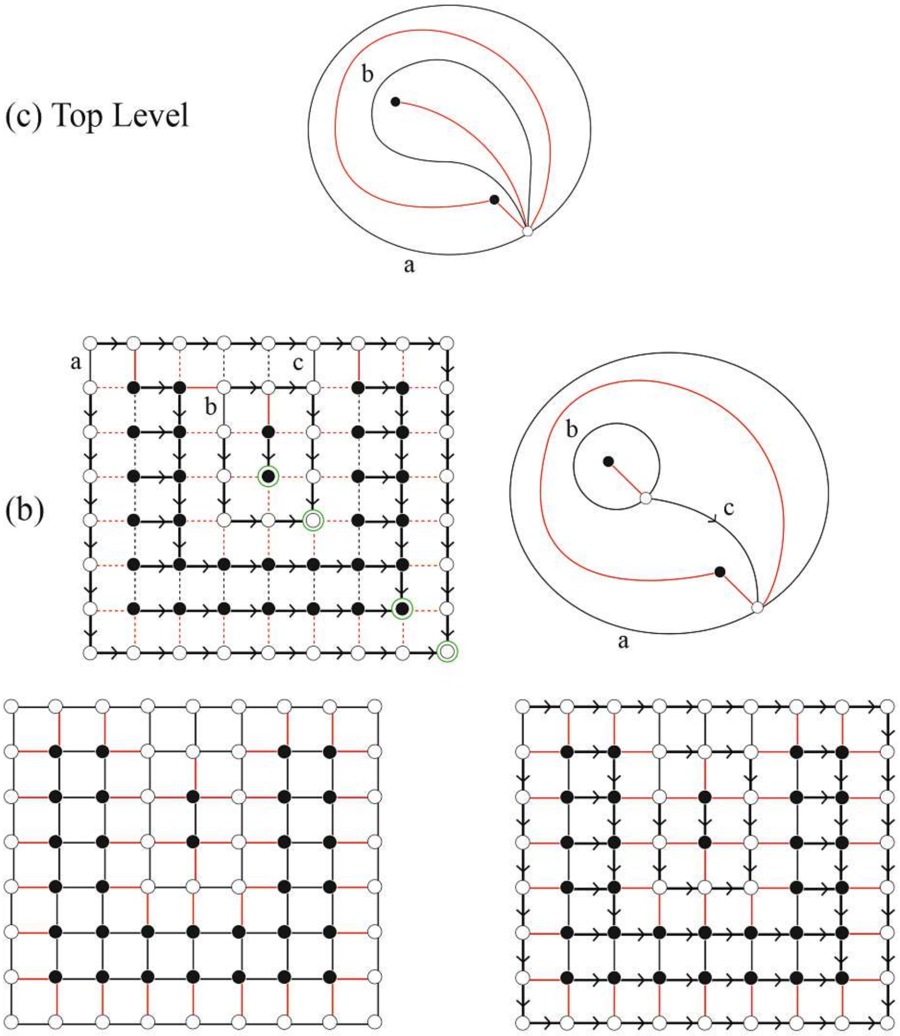


Figure 3.4: An example of the usual construction [BK00] of the irregular pyramid.



(c) Top Level

(b)

(a) Base Level

Figure 3.5: Construction of the irregular pyramid using the RtC method. The input graph at the base level is identical to Fig. 3.4.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

CHAPTER 4

Parallel Algorithms

> *Alone we can do so little; together we can do so much.*
>
> ***Helen Keller***

*This chapter highlights the advantages of designing parallel algorithms proposed in **All Papers** of the cumulative thesis, presented in Section 4.1. Section 4.2 defines independent edges and independent darts. The discussion then briefly covers the two main operations in the irregular pyramid, namely parallel edge removal and parallel edge contraction (Sections 4.3 and 4.4).*

4.1 Parallel Algorithms

In the era of big data [Kit14], characterized by an explosion of data from various sources such as social media [MBD+12], machine learning [WZWD13], artificial intelligence [OE16], and healthcare [MSC13], the importance of parallel algorithms has become increasingly critical. This surge in data, further amplified by fields like bioinformatics, astronomy, and climate science, has made the data processing tasks exponentially more complex [NJDM+22].

Parallel algorithms address this challenge by offering enhanced efficiency and scalability. They leverage multiple processors to expedite computation and accommodate growing data volumes, an approach that traditional serial algorithms cannot match due to the vast scale of modern data sets.

The goal of parallel algorithms particularly in this thesis is to **design algorithms** that can perform multiple computations simultaneously. Parallel algorithms take a problem and divide it into discrete, independent parts so that each part can be executed concurrently, typically on separate processing cores or computers [CLRS22]. This concurrent execution is a key aspect that differentiates parallel algorithms from traditional serial algorithms, which solve one part of the problem at a time.

Hierarchical structures are particularly useful for parallel algorithms. By defining *independent elements* within irregular pyramids, local computations at each level of the hierarchy can be executed independently. This characteristic is especially beneficial in a parallel computing context where independent tasks can be allocated to various processors for concurrent execution. Furthermore, each level of the pyramid can be processed in parallel, significantly enhancing computational efficiency. Additionally, different levels could be processed on distinct hardware platforms, each optimized for the scale of data it handles.

4.2 Independent Elements

In the realm of parallel algorithms, independent elements refer to portions of data or problem segments that can be processed in parallel, without interdependence on the outcomes of other segments [RR23]. This independence allows for simultaneous processing, eliminating the need for data exchange between elements. In the specific case of constructing irregular pyramids, the process involves edge removal and edge contraction operations. The key to efficient parallel processing in this context is the identification of *independent edges*, which can be processed concurrently for each operation.

4.2.1 Independent Edges

Definition 14 (Independent Edges)**.** *In a connected plane graph two edges not sharing an endpoint are independent edges.*

In case the irregular pyramid uses the combinatorial map structure instead of a neighborhood graph, the fundamental elements are *darts* where each dart d is a half-edge. In this manner, an edge is denoted by $e = (d, \alpha(d))$. The edge removal and edge contraction in the combinatorial pyramid modifies incident darts to the edge e.

Definition 15 (Dependent Darts)**.** *In a combinatorial map a set of four darts defines dependent darts (DD) of an edge* $e = (d, \alpha(d))$ *such that:*

$$DD(d, \alpha(d)) = \{\sigma(d), \sigma^{-1}(d), \sigma(\alpha(d)), \sigma^{-1}(\alpha(d))\} \tag{4.1}$$

Therefore, the independent darts are defined as follows:

Definition 16 (Independent Darts)**.** *Two darts* d_1 *and* d_2 *are independent iff*

$$DD(d_1, \alpha(d_1)) \cap DD(d_2, \alpha(d_2)) = \emptyset \tag{4.2}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Independent edges in the neighborhood graph, or independent darts in the combinatorial map, set the stage for the pyramid's parallel operations. Fig. 4.1 illustrates both independent edges and independent darts. While two edges that share a vertex are always dependent, two darts that share a vertex may become independent, as shown in Fig. 4.1-right.

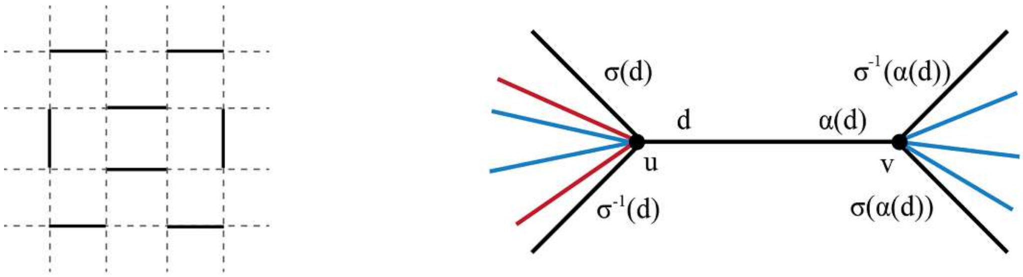


Figure 4.1: Left: a set of independent edges and Right: a set of dependent darts (black color). Two red darts are independent.

4.3 Parallel Edge Removal

Paper B introduces a method for parallel edge removal in a $2D$ grid structure of an image's neighborhood graph. By identifying sets of non-sharing endpoints in grid rows and columns, independent edges are recognized in four systematic steps, enabling the removal of all redundant edges with parallel constant complexity. This process ensures efficient edge elimination across the connected component in just four steps. Fig. 4.2 demonstrates the encoding of independent edges in a grid structure example.

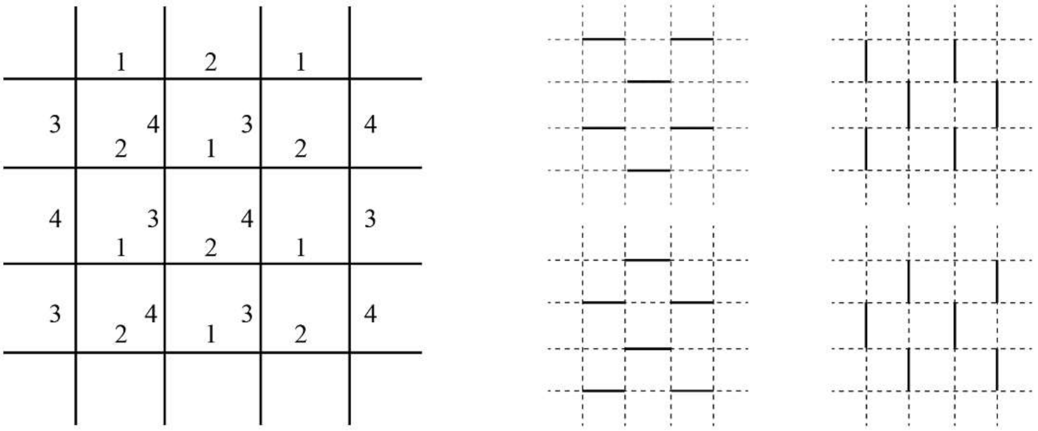


Figure 4.2: Four distinct sets of independent edges.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

4.4 Parallel Edge Contraction

Contrary to traditional methods that randomly select contraction kernels (CKs) at each level of the irregular pyramid [Mee89a, KHPL05], the proposed method employs a total order-based approach to determine all CKs in just two steps. Algorithm 3 details the Fast Labeled Spanning Tree (FLST) method introduced in Paper D, highlighting that Steps 4 and 12 are crucial for selecting all CKs. The outcome is a binary irregular pyramid, where each top-level surviving vertex corresponds to a spanning tree that represents its connected component within the receptive field.

Algorithm 3 Fast Labeled Spanning Tree (FLST)

```
 1: Input: A 4-connected neighborhood graph $G = (V, E)$ of a binary input image
 2: Initialization: Establish the total vertex order according to equations (1) and (2)
    in Paper A and set $K = 0$, where $K$ denotes the level of the irregular pyramid.
 3: $E = E_0 \dot{\cup} E_1$
    {First step of contraction kernels selection}
 4: Select contraction kernels, $S_1$, from $E_0$ (equation (4) in Paper A)
 5: Remove redundant edges (using Algorithm 1, Steps 5 to 8)
 6: while there is an edge in $S_1$ do
 7:    Identify the independent set of edges, $S'$, from $S_1$
 8:    Contract the edges of $S'$
 9:    $S_1 \leftarrow S_1 \setminus S'$
10:    $K = K + 1$
11: end while
    {Second step of contraction kernels selection}
12: Select the contraction kernels, $S_2$, from the remaining zero edges
13: Remove redundant edges (using Algorithm 1, Steps 5 to 8)
14: while there is an edge in $S_2$ do
15:    Identify the independent set of edges, $S'_2$, from $S_2$
16:    Contract the edges of $S'_2$
17:    $S_2 \leftarrow S_2 \setminus S'_2$
18:    $K = K + 1$
19: end while
20: Output: An irregular pyramid
```

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

CHAPTER 5

Fundamental Operations in Binary Images

> *Making the simple complicated is commonplace; making the complicated simple, awesomely simple, that's creativity.*
>
> ***Charles Mingus***

*This chapter begins by elucidating local and global operations in binary images in Section 5.1. In Section 5.2, **Paper B** is referenced, and a brief overview is provided on how connected component labeling is performed using irregular pyramids. Section 5.3 refers to **Papers C** and **E**, delving into the investigation of computing the distance transform with parallel logarithmic complexity. Furthermore, computing the DT in connected plane graphs, n-Gmaps, and combinatorial maps is also briefly explained.*

5.1 Operations in Binary Images

Binary image processing is crucial for analyzing and interpreting image content, involving operations like thresholding, dilation, erosion, opening, closing, connected component labeling (CCL), and distance transform (DT), which enable a variety of applications [Sb85, AB94].

Operations in image analysis are categorized into local and global processes. **Local processes**, such as dilation and erosion, affect a pixel based on its immediate neighbors [BCR90], while **global processes** like CCL and DT depend on a broader set of pixels, potentially the entire image, to determine a pixel's value [Ros83]. Global operations, computed through multiple passes or algorithms, leverage wide spatial relationships to extract further insights from the image, despite their reliance on local information.

Utilizing irregular graph pyramids to encode both local and global information of binary images, Paper B presents a new method for computing CCL, while Paper C and D introduce novel algorithms for computing the DT.

5.2 Connected Component Labeling

Connected Component Labeling (CCL) is a fundamental process in binary image analysis that labels connected groups of pixels, enabling the distinction of separate objects or features [SHS03]. In binary images, this involves identifying and labeling pixel groups based on their foreground or background status, facilitating the separate analysis of each object [HRG+17].

CCL algorithms are divided into **sequential** and **parallel** categories. Sequential methods, like the two-pass or one-pass algorithms, process pixels in a specific sequence, suitable for serial computation [HCS11]. Parallel algorithms, in contrast, handle multiple pixels at once, benefiting from parallel processing hardware like GPUs for enhanced efficiency [HRG+17].

Both sequential and parallel CCL algorithms generally exhibit linear computational complexity, $O(n)$, where n is the number of pixels in the image, although parallel approaches can leverage modern hardware to improve processing time [WCC+03, HCSW09].

Paper B proposes a parallel method named //ACC[1] that not only offers a parallel algorithm for CCL, even in the worst-case scenario but also establishes topological relations between connected components. Alg. 4 outlines the //ACC method.

Algorithm 4 Parallel Pyramidal Connected Component (//ACC)

```
Input: A 4-connected neighborhood graph $G = (V, E)$ of a binary input image, $G_K = (V_K, E_K)$: a simplified, smaller graph at level $K$, where $V_K$ and $E_K$ are the vertices and edges at this level, respectively.
Construct the irregular pyramid (using Algorithm 3)
Assign a unique label to each surviving vertex at the top level of the irregular pyramid
{Begin top-down propagation of labels}
while $K > 1$ do
    Each vertex at level $K$ propagates its label to its corresponding children at level $K - 1$
    $K = K - 1$
end while
Output: Connected component labeling (CCL) of the input graph
```

---

[1] It is pronounced pac where the // and A stand for parallel and pyramidal.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

5.3 Distance Transform

The distance transform determines the minimum distance of each pixel in an image from the nearest obstacle, using distance measures like Euclidean, Manhattan, or chessboard [Bor84]. In binary images, it calculates the distance from background pixels to the nearest foreground pixel, producing a grayscale image where pixel intensities reflect these distances.

The computational complexity of computing distance transforms in common methods [FCTB08], which only involve local processing, is **linear**. This includes parallel algorithms [CTMT10, ELRTBZ^{+}20] that, despite employing concurrent operations, still exhibit linear propagation between processing elements. However, Paper C and Paper E introduce methods that achieve faster distance propagation.

5.3.1 Computation of Distance Transform Utilizing Irregular Pyramids

The goal in computing the Distance Transform (DT) is to measure the distance of each foreground pixel from the background, starting from boundary pixels (seeds) and extending through the foreground's connected components. Paper C leverages the //ACC method from Paper B for efficient distance computation by identifying seeds at vertices incident to inter-CCs edges. The construction of the irregular pyramid and selection of independent contraction kernels enable distance propagation with **power-of-two** numbers, shifting the complexity from linear to parallel logarithmic.

5.3.2 Geodesic Distance Transform

The Geodesic Distance Transform (GDT), unlike the standard Distance Transform, computes distances from each pixel to specific constraint points or regions within a defined area or based on certain constraints. Paper C presents a method for GDT computation that primarily differs from DT in the initialization of seeds, which are selected points of interest in the foreground. The aim is to calculate the minimum distance from each foreground point to these seeds, focusing on the *shortest path* within permissible routes of a given structure or shape. Algorithm 5 details the procedure for the computation of the GDT.

5.3.3 DT in Connected Plane Graphs

The Distance Transform (DT) is adaptable to connected planar graphs beyond grid structures by establishing a total order over vertices to form a spanning tree through selected edges. Paper C proposes using the *breadth-first search* (BFS) for distance propagation across this tree, achieving parallel complexity of $O(\delta(T))$, with $\delta(T)$ being the longest path in the foreground's spanning forest. Banaeyan and Kropatsch [BK24] have further refined this to reduce worst-case linear complexity, $O(n)$, to parallel logarithmic complexity.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Algorithm 5 Computing the Geodesic Distance Transform (GDT) via the Irregular Pyramid

```
Input: A 4-connected neighborhood graph $G = (V, E)$ of a binary input image, the foreground object: $F$, a set of seeds: $S$, $S \subset F$
Initialization: Establish the total vertex order according to equations (1) and (2) in Paper A and set $K = 0$, where $K$ denotes the level of the irregular pyramid. Set $DT(s) = 0$ for all $s \in S$, and $DT(v) = \infty$ for all $v \in F \setminus S$
$E = E_0 \mathbin{\dot\cup} E_1$
while there exists a vertex at level $K$ such that $DT(v) = \infty$ do
    Identify independent contraction kernels from zero-edges that have $\infty$ distance at both endpoints
    Contract the identified independent contraction kernels
    Record the number of contractions ($\boxed{i}$) for each surviving edge
    Propagate distances ($2^{\boxed{i}}$) from vertices with calculated distance to their neighboring vertices
    $K = K + 1$
end while{Top of the pyramid is reached}
while $K > 0$ do
    $K = K - 1$
    Inherit computed distances from the level above
    Propagate distances ($2^{\boxed{i}}$) from vertices with calculated distance to their neighboring vertices
end while{Base of the pyramid is reached}
Output: Geodesic DT for each vertex in the connected component of interest
```

5.3.4 DT in n-Gmaps and Combinatorial Maps

Paper C presents a novel method for computing the Distance Transform (DT) within *n-dimensional generalized maps* (n-Gmap) [Lie91, DL14, BKH22], using *darts* (or half edges) as foundational elements. Initialization can involve single darts, a single *i-cell* $(0 < i \leq n)$, or combinations thereof. This approach generalizes the graph-based propagation method, utilizing involutions and darts for smoother, more precise DT computation across n-Gmaps or combinatorial maps, as discussed in Papers C and E. It allows distance propagation around i-cell boundaries, defining n distinct distances that provide insights into various dimensions.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

CHAPTER 6

Experiments and Results

An experiment is a question which science poses to Nature, and a measurement is the recording of Nature's answer.

Max Planck

This chapter presents the experiments and results of the papers included in the cumulative thesis. The details of the results, including resulting tables and comparison diagrams, are discussed in Papers A, B, and E.

6.1 Initialization and Pre-processing in Proposed Algorithms

Algorithms 1-5 presented in prior sections presuppose knowledge of the binary input image's size. These algorithms utilize the combinatorial map structure for pyramid construction, with darts—each representing a half-edge—as the primary elements. The implementation of the combinatorial map [BK12] employs canonical encoding [TK14b] to maintain the σ-permutation of each dart within a 1D array. Given the input image size, the arrangement of darts in this encoding, and consequently the independent darts at the pyramid's base level, are predefined. Thus, the priority for their contraction (as outlined in the logarithmic encoding proposed in Paper B) is calculated in advance of initiating the algorithms.

6.2 Redundant Edge Elimination

The *RtC* method, as applied in Paper B across a diverse set of binary images, has shown that up to 50% of edges can be redundant, with empirical evidence suggesting an average redundancy of over 45%. This efficiency enables a significant reduction in memory usage—up to 45% less—when constructing the irregular pyramid using the combinatorial map structure and the canonical ordering.

6.3 Connected Component Labeling

Paper B's //ACC method, when applied to various image types, shows varied performance: it is slightly slower for small images but exhibits improved results for larger ones. Unique to //ACC is the provision of topological information between connected components, a feature not commonly available in standard CCL methods, which adds value beyond mere labeling efficiency.

6.4 Distance Transform

Paper E's comparison of DT computation using irregular pyramids against traditional methods highlights significant speed improvements, especially for Random images with smaller objects. Additionally, Papers E and C enhance the DT computation in combinatorial maps and n-Gmaps, offering smoother results and introducing n distinct distances for multidimensional analysis. This approach has practical applications, such as simulating gas exchange in leaves, illustrating the method's utility in real-world scenarios.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

CHAPTER 7

Concluding Remarks

In this concluding chapter, the original contributions of the thesis are summarized, and potential avenues for future research are suggested.

7.1 Conclusion

This cumulative thesis introduces the concept of redundant edges and their structures within a 2D binary irregular graph pyramid. In Papers A and C, the pyramid's edges are categorized into intra-CC and inter-CCs edges. Through a parallel approach, all redundant intra-CC edges at the base level are effectively eliminated. Moreover, a notable proportion of inter-CCs edges also undergo elimination at the base level.

Paper B introduces the Remove then Contract (RtC) method to remove redundant edges, offering several advantages. Firstly, the RtC method reduces the complexity of removing redundant edges from sequential linear to parallel constant complexity. Secondly, it has the potential to decrease the memory requirements for pyramid construction. Theoretical findings presented in Paper A show that the upper bound of this reduction can reach 50%. However, specific cases, such as checkerboard patterns containing no redundant edges, may be exempt from this reduction. Empirical evidence demonstrates that the average proportion of redundant edges across various classes of binary images falls within the range of 45% to 49%.

The thesis leverages the parallel complexity of constructing the irregular pyramid to significantly reduce the computational complexity of fundamental operations in binary images. The transition is from sequential linear complexity $O(n)$ to parallel logarithmic complexity $O(\log n)$, where n denotes the number of pixels/vertices in the input binary image/graph. Of particular focus are the connected component labeling (CCL) and distance transform (DT) operations, which require both global and local processes. The parallel pyramidal connected component labeling (//ACC) algorithm proposed in Paper B not only achieves efficient CCL but also preserves crucial topological information, including inclusion relationships between objects and multi-boundaries, which conventional CCL methods tend to overlook. While the proposed method may execute slower for small-sized images (Paper B), it demonstrates accelerated performance as the image size increases (Paper B, D and E).

Paper C and E present innovative algorithms designed to compute the distance transform (DT). These algorithms utilize a set of power-of-two numbers to propagate distances within a connected component, departing from the traditional one-step propagation method. With parallel logarithmic complexity, these algorithms significantly enhance execution time. Moreover, Paper C introduces a new DT on the generalized map, resulting in a more precise and smoother DT representation of the input image. Notably, the $n-Gmap$ incorporates n different distances, a feature particularly useful for biologists seeking to simulate $CO2$ infusions within the 2D air space of a leaf or along the 1D boundary of the leaf's cells.

Last but not least, the algorithms proposed in all papers (Papers A, B, C, D, and E) demonstrate parallel complexity and benefit from a sufficient number of independent processing elements for optimal performance. As the number of processing elements increases, the efficiency of these parallel algorithms becomes even more pronounced.

7.2 Future Work

This thesis introduces a method for extracting redundant edges in binary images, a concept that holds potential for extension to gray-scale and color images. For gray-scale images, connected components can possess any gray value, yet the concept of intra-CC (Connected Component) edges remains applicable. Redundant intra-CC edges can be identified using a similar approach as with binary images. By adjusting the definition of inter-CCs edges to include all edges with a contrast value greater than zero, and subsequently removing redundant intra-CC edges, the resulting graph's inter-CCs edges with the minimum contrast value can be reclassified as intra-CC edges. This process allows for the detection of redundant edges in gray-scale images through iterative selection and reclassification of edges, potentially completing in a maximum of m steps, where m represents the number of distinct contrast values in the image's neighborhood graph.

For RGB color images, an approach could involve constructing three separate gray-scale pyramids from the red, green, and blue channels. Each pyramid could then be processed similarly to the gray-scale image method, enabling the extraction of redundant edges in color images by treating each color channel independently.

By utilizing combinatorial maps or n-generalized maps for constructing the irregular pyramid, the algorithm can be extended to higher dimensions, transitioning from 2D to 3D or even nD scenarios. This extension could significantly broaden the range of potential applications.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

This thesis introduced a total ordering of vertices to create the spanning tree or Equivalent Contraction Kernel (ECK). An intriguing inverse problem emerges from this: Given a pre-determined spanning tree for a connected component, what would constitute an optimal valid total order? The spanning tree could be provided based on the specific properties of objects in the input image. For instance, if one aims to preserve the mass center of connected components, what would be the appropriate total order?

The algorithms proposed in this thesis for Connected Component Labeling (CCL) and Distance Transform (DT) could be extended to generally connected plane graphs (non-grid structures) [BK24]. This extension could benefit larger communities that are not limited to working with array data, broadening the applicability of these methods.

The idea of computing the distance transform can be further expanded to calculate the distance map in a given cellular complex. Depending on the properties one wishes to preserve, the suitable distance transform can be defined.

Lastly, utilizing the hierarchical structure could assist us in reducing the complexity of the eccentricity transform [BK23b]. This is a special case of the distance transform method, which is notably more robust against noise.

As we continue to expand these ideas, we can anticipate that they will make substantial contributions to various fields of study, not just in image processing but also in data analysis and computational geometry, among others. These extensions, therefore, hold great promise for the future of pyramid-based methods in image and data analysis.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Bibliography

[AB94] Jim Austin and Stephen Buckle. The practical application of binary neural networks. In *Proc. of the UNICOM seminar on Adaptive computing and information processing*. Citeseer, 1994.

[BBK22] Majid Banaeyan, Darshan Batavia, and Walter G. Kropatsch. Removing redundancies in binary images. In *International Conference on Intelligent Systems and Patterns Recognition (ISPR), Hammamet, Tunisia, March 24-25, 2022*, page 221–233. Springer, 2022.

[BCKH22] Majid Banaeyan, Carmine Carratù, Walter G. Kropatsch, and Jiří Hladůvka. Fast distance transforms in graphs and in gmaps. In *IAPR Joint International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), Montreal, Canada, August 26-27, 2022*, pages 193–202. Springer, Lecture Notes in Computer Science,volume 13813, 2022.

[BCR90] Michel Bister, Jan Cornelis, and Azriel Rosenfeld. A critical view of pyramid segmentation algorithms. *Pattern Recognition Letters*, 11(9):605–617, 1990.

[BHR81] Peter J Burt, Tsai-Hong Hong, and Azriel Rosenfeld. Segmentation and estimation of image region properties through cooperative hierarchial computation. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(12):802–809, 1981.

[BK00] Luc Brun and Walter G. Kropatsch. Irregular pyramids with combinatorial maps. In Francesc J. Ferri, José M. Iñesta, Adnan Amin, and Pavel Pudil, editors, *Advances in Pattern Recognition, Joint IAPR International Workshops on SSPR'2000 and SPR'2000*, volume 1876, pages 256–265. Springer, Berlin Heidelberg, New York, 2000.

[BK01] Luc Brun and Walter Kropatsch. Introduction to combinatorial pyramids. In *Digital and image geometry*, pages 108–128. Springer, 2001.

[BK12] Luc Brun and Walter G. Kropatsch. Hierarchical graph encodings. In Olivier Lézoray and Leo Grady, editors, *Image Processing and Analysis with Graphs: Theory and Practice*, pages 305–349. CRC Press, 2012.

[BK22a] Majid Banaeyan and Walter G. Kropatsch. Fast labeled spanning tree in binary irregular graph pyramids. *Journal of Engineering Research and Sciences, Volume 1, Issue 10*, pages 69–78, 2022.

[BK22b] Majid Banaeyan and Walter G. Kropatsch. Parallel o(log(n)) computation of the adjacency of connected components. In *3rd International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI), Paris, France, June 1-3, 2022*, pages 102–113. Springer, Lecture Notes in Computer Science,volume 13364, 2022.

[BK23a] Majid Banaeyan and Walter G. Kropatsch. Distance transform in parallel logarithmic complexity. In *Proceedings of the 12th International Conference on Pattern Recognition Applications and Methods - ICPRAM*, pages 115–123. SciTePress, 2023.

[BK23b] Majid Banaeyan and Walter G. Kropatsch. Reducing the computational complexity of the eccentricity transform of a tree. In *Proceedings of the 13th AIPR-TC15 International Workshop on Graph Based Representation*, pages 160–171. Springer, Lecture Notes in Computer Science,volume 14121, 2023.

[BK24] Majid Banaeyan and Walter G. Kropatsch. Distance transform in images and connected plane graphs. In *International Conference on Pattern Recognition Applications and Methods*, pages 19–32. Springer, Lecture Notes in Computer Science,volume 14547, 2024.

[BKH22] Majid Banaeyan, Walter G. Kropatsch, and Jiří Hladuvka. Redundant 1-cells in multi-labeled 2-gmap irregular pyramids. pages 5–11, 2022.

[Bor84] Gunilla Borgefors. Distance transformations in arbitrary dimensions. *Computer vision, graphics, and image processing*, 27(3):321–345, 1984.

[BR14] Antony Bryant and Uzma Raja. In the realm of big data. *First monday*, 19(2), 2014.

[Chr75] Nicos Christofides. *Graph theory: An algorithmic approach (Computer science and applied mathematics)*. Academic Press, Inc., 1975.

[CJGK16] Martin Cerman, Ines Janusch, Rocio Gonzalez-Diaz, and Walter G. Kropatsch. Topology-based image segmentation using LBP pyramids. *Machine Vision and Applications*, 27(8):1161–1174, 2016.

[CLRS22] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

[CSKH90] Christine A Curcio, Kenneth R Sloan, Robert E Kalina, and Anita E Hendrickson. Human photoreceptor topography. *Journal of comparative neurology*, 292(4):497–523, 1990.

[CTMT10] Thanh-Tung Cao, Ke Tang, Anis Mohamed, and Tiow-Seng Tan. Parallel banding algorithm to compute exact distance transform with the gpu. In *Proceedings of the 2010 ACM SIGGRAPH symposium on Interactive 3D Graphics and Games*, pages 83–90, 2010.

[DK18] Anand Deshpande and Manish Kumar. *Artificial intelligence for big data: Complete guide to automating big data solutions using artificial intelligence techniques*. Packt Publishing Ltd, 2018.

[DL14] Guillaume Damiand and Pascal Lienhardt. *Combinatorial maps: efficient data structures for computer graphics and image processing*. CRC Press, 2014.

[Duf86] Michael JB Duff. Pyramids—expected performance. In *Pyramidal Systems for Computer Vision*, pages 59–73. Springer, 1986.

[ELRTBZ$^+$20] Juan Carlos Elizondo-Leal, José Gabriel Ramirez-Torres, Jose Hugo Barrón-Zambrano, Alan Diaz-Manríquez, Marco Aurelio Nuño-Maganda, and Vicente Paul Saldivar-Alonso. Parallel raster scan for euclidean distance transform. *Symmetry*, 12(11):1808, 2020.

[FB82] Jerome A Feldman and Dana H Ballard. Connectionist models and their properties. *Cognitive science*, 6(3):205–254, 1982.

[FCTB08] Ricardo Fabbri, Luciano Da F Costa, Julio C Torelli, and Odemir M Bruno. 2d euclidean distance transform algorithms: A comparative survey. *ACM Computing Surveys (CSUR)*, 40(1):1–44, 2008.

[Gra99] Gösta H Granlund. The complexity of vision. *Signal Processing*, 74(1):101–126, 1999.

[Hal60] Paul Richard Halmos. *Naive set theory*, pages 225–272. van Nostrand, 1960.

[Hax07] Yll Haxhimusa. *The structurally Optimal Dual Graph Pyramid and its application in image partitioning*, volume 308, pages 27–28. IOS Press, 2007.

[HCS11] Lifeng He, Yuyan Chao, and Kenji Suzuki. Two efficient label-equivalence-based connected-component labeling algorithms for 3D binary images. *IEEE Transactions on Image Processing*, 20(8):2122–2134, 2011.

[HCSW09] Lifeng He, Yuyan Chao, Kenji Suzuki, and Kesheng Wu. Fast connected-component labeling. *Pattern recognition*, 42(9):1977–1987, 2009.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

[HGK03] Yll Haxhimusa, Roland Glantz, and Walter G. Kropatsch. Constructing stochastic pyramids by mides - maximal independent directed edge set. In Edwin Hancock and Mario Vento, editors, *4th IAPR-TC15 Workshop on Graph-based Representation in Pattern Recognition*, volume 2726, pages 24–34. Springer, Berlin Heidelberg, New York, 2003.

[HGS$^+$02] Yll Haxhimusa, Roland Glantz, Maamar Saib, Georg Langs, and Walter G. Kropatsch. Logarithmic tapering graph pyramid. In *Joint Pattern Recognition Symposium*, pages 117–124. Springer, 2002.

[HIK06] Yll Haxhimusa, Adrian Ion, and Walter G. Kropatsch. Irregular pyramid segmentations with stochastic graph decimation strategies. In José Francisco Martínez-Trinidad, Jesús Ariel Carrasco Ochoa, and Josef Kittler, editors, *Progress in Pattern Recognition, Image Analysis and Applications*, volume LNCS 4225, pages 277–286. Springer, Berlin Heidelberg, 2006.

[Hil16] Martin Hilbert. Big data for development: A review of promises and challenges. *Development Policy Review*, 34(1):135–174, 2016.

[HRG$^+$17] Lifeng He, Xiwei Ren, Qihang Gao, Xiao Zhao, Bin Yao, and Yuyan Chao. The connected-component labeling problem: A review of state-of-the-art algorithms. *Pattern Recognition*, 70:25–43, 2017.

[HSGK01] Yll Haxhimusa, Maamar Saib, Roland Glantz, and Walter G. Kropatsch. Equivalent contraction kernels using dynamic trees. In Bov stjan Likar, editor, *Computer Vision - 6th CVWW, Computer Vision Winter Workshop*, pages pp. 267–275. Slovenian Pattern Recognition Society, 2001.

[JM92] Jean-Michel Jolion and Annick Montanvert. The adaptive pyramid: a framework for 2d image analysis. *CVGIP: Image Understanding*, 55(3):339–348, 1992.

[JR12] Jean-Michel Jolion and Azriel Rosenfeld. *A pyramid framework for early vision: multiresolutional computer vision*, volume 251, pages 6–18. Springer Science & Business Media, 2012.

[KB24] Walter G. Kropatsch and Majid Banaeyan. *Controlling Topology Preserving Graph Pyramids*. Pattern Recognition and Intelligent Systems. World Scientific book, 2024.

[KHPL05] Walter G. Kropatsch, Yll Haxhimusa, Zygmunt Pizlo, and Georg Langs. Vision Pyramids that do not Grow too High. *Pattern Recognition Letters*, Vol. 26(3):319–337, 2005.

[Kit14] Rob Kitchin. Big data, new epistemologies and paradigm shifts. *Big data and society*, 1(1), 2014.

TU WIEN Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

[Kle14] Reinhard Klette. *Concise computer vision*, volume 233, pages 90–94. Springer, 2014.

[Kro87] Walter G. Kropatsch. Curve representations in multiple resolutions. *Pattern Recognition Letters*, 6(3):179–184, 1987.

[Kro95] Walter G. Kropatsch. Building Irregular Pyramids by Dual Graph Contraction. *IEE-Proc. Vision, Image and Signal Processing*, Vol. 142(No. 6):pp. 366–374, December 1995.

[Kro98] Walter G. Kropatsch. Equivalent weighting functions to equivalent contraction kernels. In *Sixth International Workshop on Digital Image Processing and Computer Graphics: Applications in Humanities and Natural Sciences*, volume 3346, pages 310–320. International Society for Optics and Photonics, 1998.

[Kro02] Walter G. Kropatsch. Abstraction pyramids on discrete representations. In *International Conference on Discrete Geometry for Computer Imagery*, pages 1–21. Springer, 2002.

[Lie91] Pascal Lienhardt. Topological models for boundary representation: a comparison with n-dimensional generalized maps. *Computer-aided design*, 23(1):59–82, 1991.

[MBD$^+$12] Andrew McAfee, Erik Brynjolfsson, Thomas H Davenport, DJ Patil, and Dominic Barton. Big data: the management revolution. *Harvard business review*, 90(10):60–68, 2012.

[Mee89a] Peter Meer. Stochastic image pyramids. *Computer Vision, Graphics, and Image Processing*, 45(3):269–294, 1989.

[Mee89b] Peter Meer. Stochastic image pyramids. *Computer Vision, Graphics, and Image Processing*, Vol. 45(No. 3):269–294, March 1989.

[MMR91] Annick Montanvert, Peter Meer, and Azriel Rosenfeld. Hierarchical image analysis using irregular tessellations. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (4):307–316, 1991.

[MSC13] Viktor Mayer-Schönberger and Kenneth Cukier. *Big data: A revolution that will transform how we live, work, and think*. Houghton Mifflin Harcourt, 2013.

[NJDM$^+$22] Muhammad Naeem, Tauseef Jamal, Jorge Diaz-Martinez, Shariq Aziz Butt, Nicolo Montesano, Muhammad Imran Tariq, Emiro De-la Hoz-Franco, and Ethel De-La-Hoz-Valdiris. Trends and future perspective challenges in big data. In *Advances in Intelligent Data Analysis and Applications: Proceeding of the Sixth Euro-China Conference on Intelligent*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Data Analysis and Applications, 15–18 October 2019, Arad, Romania, pages 309–325. Springer, 2022.

[OE16] Ziad Obermeyer and Ezekiel J Emanuel. Predicting the future—big data, machine learning, and clinical medicine. *The New England journal of medicine*, 375(13):1216, 2016.

[Piz22] Zygmunt Pizlo. *Problem Solving, Cognitive Mechanisms and Formal Models*. Cambridge University Press, 2022.

[PVT+22] Alexander Palmrich, Klara Voggeneder, Danny Tholen, Guillaume Theroux-Rancourt, " JH ", and Walter Kropatsch. An unsupervised, shape-based 3D cell instance segmentation method for plant tissues. In *OAGM Workshop: Digitalization for Smart Farming and Forestry*, pages 54–59, 2022.

[PWHM14] Mary C Potter, Brad Wyble, Carl Erick Hagmann, and Emily S McCourt. Detecting meaning in rsvp at 13 ms per picture. *Attention, Perception, & Psychophysics*, 76(2):270–279, 2014.

[Ros83] Azriel Rosenfeld. Quadtrees and pyramids: Hierarchical representation of images. *Pictorial data analysis*, pages 29–42, 1983.

[Ros86] Azriel Rosenfeld. Some pyramid techniques for image segmentation. In *Pyramidal Systems for Computer Vision*, pages 261–271. Springer, 1986.

[RR23] Thomas Rauber and Gudula Rünger. *Parallel Programming Models*, pages 105–167. Springer International Publishing, 2023.

[Sb85] Satoshi Suzuki and KeiichiA be. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1):32–46, 1985.

[SHS03] Kenji Suzuki, Isao Horiba, and Noboru Sugie. Linear-time connected-component labeling based on sequential local operations. *Computer Vision and Image Understanding*, 89(1):1–23, 2003.

[SMK10] Dan Shao, LA Mateos, and WG Kropatsch. Irregular laplacian graph pyramid. In *Computer Vision Winter Workshop 2010 (Nove Hrady, Czech Republic: Czech Pattern Recognition Society)*, 2010.

[TK14a] Fuensanta Torres and Walter G. Kropatsch. Canonical Encoding of the Combinatorial Pyramid. In Zuzana Kúkelová and Jan Heller, editors, *Proceedings of the 19th Computer Vision Winter Workshop 2014*, pages 118–125, Křtiny, CZ, February 2014. ISBN: 978-80-260-5641-6.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

[TK14b] Fuensanta Torres and Walter G. Kropatsch. Canonical encoding of the combinatorial pyramid. In *Proceedings of the 19th Computer Vision Winter Workshop*, pages 118–125, 2014.

[Tru93] R.J. Trudeau. *Introduction to Graph Theory*, page 64. Dover Books on Mathematics. Dover Pub., 1993.

[Uhr86] Leonard Uhr. Parallel, hierarchical software/hardware pyramid architectures. In Virginio Cantoni and Stefano Levialdi, editors, *Pyramidal Systems for Image Processing and Computer Vision*, volume F25 of *NATO ASI Series*, pages 1–20. Springer-Verlag Berlin, Heidelberg, 1986.

[Uhr87] Leonard Uhr. Highly parallel, hierarchical, recognition cone perceptual structures. *Parallel computer vision*, 4:249–292, 1987.

[Uhr14] Leonard Uhr. *Parallel computer vision*. Elsevier, 2014.

[WBT+19] Yiyi Wang, Nicolas Bensaid, Pavan Tiruveedhula, Jianqiang Ma, Sowmya Ravikumar, and Austin Roorda. Human foveal cone photoreceptor topography and its dependence on eye length. *Elife*, 8:e47148, 2019.

[WCC+03] Kuang-Bor Wang, Tsorng-Lin Chia, Zen Chen, Der-Chyuan Lou, et al. Parallel execution of a connected component labeling operation on a linear array architecture. *Journal of Information science and engineering*, 19(2):353–370, 2003.

[Wil95] Dieter Willersinn. *Irreguläre Kurvenpyramiden: ein Schema für perzeptuelle Organisation*. PhD thesis, Vienna University of Technology, 1995. PhD.

[WZWD13] Xindong Wu, Xingquan Zhu, Gong-Qing Wu, and Wei Ding. Data mining with big data. *IEEE transactions on knowledge and data engineering*, 26(1):97–107, 2013.

[Zek93] S Zeki. *A vision of the brain Blackwell*. Oxford, England, 1993.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

TU WIEN Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Second Part

TU WIEN Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Paper A
Removing Redundancies in Binary Images

Reprinted from [BBK22] with kind permission from Springer Nature.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Removing Redundancies in Binary Images

Majid Banaeyan( ), Darshan Batavia, and Walter G. Kropatsch

Pattern Recognition and Image Processing Group 193/03, TU Wien, Vienna, Austria
{majid,darshan,krw}@prip.tuwien.ac.at

Abstract. Every day a huge amount of digital data is generated. Processing such big data encourages efficient data structure and parallelized operations. In this regard, this paper proposes a graph-based method reducing the memory requirement of the data storage. Graphs as a versatile representative tool in intelligent systems and pattern recognition may consist of many nonessential edges accumulating memory. This paper defines the structure of such redundant edges in the neighborhood graph of a 2D binary image. We introduce a novel approach for contracting the edges that simultaneously assists in determining the structurally redundant edges. In addition, finding a set of independent edges, the redundant edges are removed in parallel with the complexity $\mathcal{O}(1)$. Theoretically, we prove that the maximum number of redundant edges is bounded by half of all edges. Practical results show the memory requirement decreases significantly depending on the input data in different categories of binary image data sets. Using the combinatorial map as the data structure, first the topological structure of the graph is preserved. Second, the method can be extended to higher dimensions (nD).

Keywords: Redundant edges · Connected component labeling · Binary image · Combinatorial map

1 Introduction

The amount of available data in intelligent systems has increased dramatically in recent years [19,20], and this situation will continue to become more extreme with the development of technologies [7,10]. Such circumstances necessitate the development of sophisticated schemes promoting better structural representation. The structure of the data helps to preserve the topology of the image and assists to achieve a compact representation of the data. This makes structure a crucial part of data analysis. The structure of data gives the information about the intrinsic relationships between the subset of the data. Helman et al. [9] stated that extraction of relevant structure helps to reduce the data storage and assists in better visualization. In their case the amount of storage required was approximately one-tenth of the actual storage required for the data. Elimination of the redundant data [18] plays a key role in achieving a compact representation of

---

Supported by Pattern Recognition and Image Processing Group (PRIP), Austria.

© Springer Nature Switzerland AG 2022
A. Bennour et al. (Eds.): ISPR 2022, CCIS 1589, pp. 221–233, 2022.
https://doi.org/10.1007/978-3-031-08277-1_19

data and saving the storage memory. Besides, it largely depends upon the representation technique, the data structure used for storage of representation, the algorithm's compliance with parallel processing, etc. This paper covers the points related to a structure preserving algorithm for binary images. More specifically we will look into the elimination of the structurally redundant data (see Sect. 3) with a graph based representation (see Sect. 2.1) using the combinatorial maps (see Sect. 2.3) as the data structure.

2 Motivations and Definitions

2.1 Graph-Based Representation

Graphs have the capabilities to represent both structured data (like images, videos, grids) as well as unstructured data (like climate data, point cloud). Narrowing down to images, graphs based representation are simple and effective. A digital image can be easily represented using a 4-adjacent neighborhood graph. Let $G = (V, E)$ be the Region Adjacency Graph (RAG) of image P where V corresponds to the vertex set and E corresponds to the edge set. The vertex $v \in V$ associates with the pixels in image P and the edge $e \in E$ connects the corresponding adjacent vertices. Let the gray-value of vertex $g(v) = g(p)$ where $p \in P$ is a pixel in the image corresponding to vertex v. Let $contrast(e)$ be an attribute of an edge $e(u, v)$ where $u, v \in V$ and $contrast(e) = |g(u) - g(v)|$. Since we are working with binary images only, the pixels (and corresponding vertices can) have either of the two values 0 and 1. Similarly the edge contrast can have only two possible values 0 and 1. The edges in the neighborhood graph can be classified into the following two categories:

Definition 1 (Zero-edge). *An edge $e \in E$ is a* ***zero-edge****, e_0, iff the contrast between its two endpoints is zero.*

Definition 2 (One-edge). *An edge $e \in E$ is a* ***one-edge****, e_1, iff the contrast between its two endpoints is one.*

The set of edges classified as e_0 is denoted as E_0 and the set of edges classified as e_1 is denoted as E_1. The edge set $E = E_0 \cup E_1$.

2.2 Image Pyramid

Image Pyramids consist of a series of successively smaller images produced from a base image. They are efficient hierarchical structures which are able to propagate local information from the base level into a global one at the top of the pyramid. Generally, two types of the pyramid, namely regular and irregular pyramid exist.

In **regular pyramids** [12] the resolution is decreased in regular steps and therefore the size of the pyramid is fixed. On the contrary, in irregular pyramids the size of the pyramid is not fixed and it is adapted to the image data. In addition, unlike the regular ones, the irregular pyramids are shift- and rotation-invariant [16] that make them useful to use in a variety of tasks, such as image

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

segmentation [6] and object recognition. It should be noticed that the irregular image pyramid is interpreted as the irregular graph pyramid when its pixels and the neighborhood relations between adjacent pixels correspond to the vertices and the edges of the graph, respectively.

Irregular Pyramids. [11,13–15] are a stack of successively reduced graphs where each graph is constructed from the graph below by selecting a specific subset of vertices and edges. For generation of irregular pyramids, we use the two fundamental operations on graphs: edge **contraction** and edge **removal** (Fig. 1). The edge contraction operation contracts an edge connecting two vertices, and the two vertices are merged into one. All edges that were incident to the merged vertices will be incident to the resulting vertex after the operation. The edge removal operation removes an edge from the graph, without changing the number of vertices or affecting the incidence relationships of other edges. In each level of the pyramid, the vertices and edges that disappear in a level above are called *non-surviving* and those that appear in the upper level *surviving* ones.

Definition 3 (Contraction Kernel (CK)). *A spanning tree of a connected component.*

A contraction kernel contracts the non-surviving vertices to their corresponding surviving vertex such that each connected component indicated by one surviving vertex.

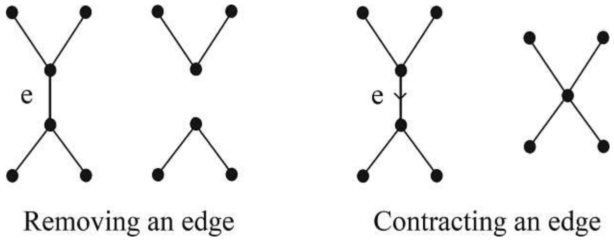


Fig. 1. Two different operations on an edge.

There are different structures to build the irregular pyramid such as simple graphs [5], dual graphs [11] and combinatorial maps (CM) [4]. In the simple graph the produced region adjacency graph (RAG) cannot distinguish between different topological configurations [13], in particular between inclusion and multiple adjacency relationships of regions [5]. The problem with dual graphs is that they cannot unambiguously represent a region enclosed in another one on a local level [5]. Therefore, in this paper the CM, as a planar embedding of RAG, is used which not only solves the mentioned problems but also provides an efficient structure that preserves topological relations between regions and can be extended to higher dimensions (nD).

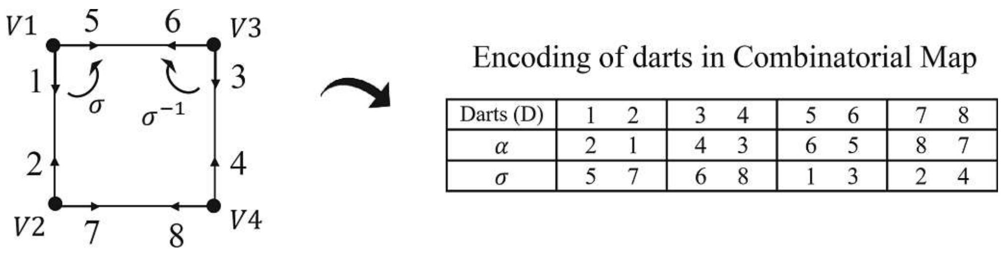
TU Bibliothek Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar. The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

A ***plane*** graph is a graph embedded in the plane such that no two edges intersect. In the plane graph there are connected spaces between edges and vertices and every such connected area of the plane is called a *face*. The *degree* of the face is the number of edges bounding the face. In addition a face bounded by a cycle is called an *empty face*. In a non-empty face traversing the boundary would require to visit vertices or edges twice.

2.3 Combinatorial Pyramid

A combinatorial pyramid is a hierarchy of successively reduced combinatorial maps. A combinatorial map (CM) is similar to a graph but explicitly stores the orientation of edges around each vertex. The combinatorial map (G) is defined by a triple $G = (D, \alpha, \sigma)$ where the D is a finite set of darts. A dart is defined as the half edge and it is the fundamental element in the CM's structure. The α is an *involution* on the set D, provides a one-to-one mapping between darts forming the same edge such that $\alpha(\alpha(d)) = d$. The σ is a *permutation* on the set D and encodes consecutive darts around the same vertex while turning counterclockwise [17]. Note that the clockwise orientation is denoted by σ^{-1}.

Figure 2 left, shows a set of adjacent darts with their σ relations in a face of degree 4. Figure 2, right, shows the encoding of the darts. For instance, consider $e = (1, 2)$ where $\alpha(1) = 2$, $\alpha(2) = 1$, $\sigma(1) = 5$.



Encoding of darts in Combinatorial Map

Darts (D)	1	2	3	4	5	6	7	8
α	2	1	4	3	6	5	8	7
σ	5	7	6	8	1	3	2	4

Fig. 2. Combinatorial map.

3 Structurally Redundant Edges

The definition of the term redundant edges differs depending on the application, the representation and the data structure used for the implementation. In our case, we are dealing with binary images. In order to obtain the structure of the binary image, the relevant edges consist of a tree that spans the connected components and the edges that interconnect the components. To detect the redundant edges, it is needed to define an efficient method for selecting the CK. Note that a connected component consists of edges with zero contrast (e_0) only, and the edges with contrast one (e_1) connect two different connected components together. Therefore, in a binary neighborhood graph, the contraction kernel is selected among only e_0s.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

3.1 Selecting the Contraction Kernel

Selecting the contraction kernel (CK) has a key role in detecting the redundant edges in the neighborhood graph. To this aim, a totally ordered set is defined over the indices of vertices. Consider the binary image has M rows and N columns such that $(1, 1)$ is the coordinate of the pixel $(p \in P)$ at the upper-left corner and (M, N) at the lower-right corner. The corresponding 4-adjacent neighborhood graph of the binary image has MN vertices. An index $Idx(., .)$ of each vertex is defined:

$$Idx : [1, M] \times [1, N] \mapsto [1, M \cdot N] \subset \mathbb{N} \tag{1}$$

$$Idx(r, c) = (c - 1) \cdot M + r \tag{2}$$

where r and c are the row and column of the pixel, respectively. Figure 3 shows the neighborhood graph of a 7 by 7 binary image where indices are from 1 to 49. Since the set of integers is totally ordered each vertex has a unique index. The important property of such totally ordered set is that every subset has exactly one minimum and one maximum member (integer number). This property provides a unique orientation between non-surviving and surviving vertices.

Consider a non-surviving vertex v. In order to find the surviving vertex, v_s, an incident e_0 must be found in its neighborhood. Such a neighborhood $\mathcal{N}(v)$ is defined as follows:

$$\mathcal{N}(v) = \{v\} \cup \{w \in V | e_0 = (v, w) \in E_0\} \tag{3}$$

if such neighborhood exists $(|\mathcal{N}(v)| > 1)$ the surviving vertex is:

$$v_s = \text{argmax}\{Idx(v_s)|\ v_s \in \mathcal{N}(v), |\mathcal{N}(v)| > 1\} \tag{4}$$

Definition 4 (Orientation of a e_0). *A $e_0 = (v, w) \in E_0$ is oriented from v to w if w has the largest index among the neighbors, $Idx(w) = \max\{Idx(u)|u \in \mathcal{N}(v)\}$. All edges to the other neighbors remain non-oriented.*

By such definition, a chain of oriented e_0s connects each non-surviving vertex to its corresponding survivor vertex. In Fig. 3 the oriented e_0s are identified by an arrow over each e_0. The three vertices (25, 33 and 49) are surviving vertices while the remaining vertices are non-surviving.

Proposition 1. *Selecting the CK partitions vertices into non-surviving and surviving vertices.*

Proof. If the $|\mathcal{N}(v)| = 1$, either there is no e_0s around v or the index of v is bigger than the indices of neighboring e_0s. Therefore the v is a surviving vertex. In case the $|\mathcal{N}(v)| > 1$, since the indices are totally ordered, there is a maximum in the neighborhood of v which is selected as survivor and the v becomes the nun-surviving vertex. □

Proposition 2. *Every non-surviving vertex has a unique surviving vertex.*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Proof. Each tree of oriented e_0s has one unique maximum as the index of the surviving vertex. ☐

Property 1. With the choice of $Idx(.)$ and the coordinate axes in (1) a non-surviving vertex contracts either to its adjacent right vertex or to its down vertex where the right vertex has the higher priority.

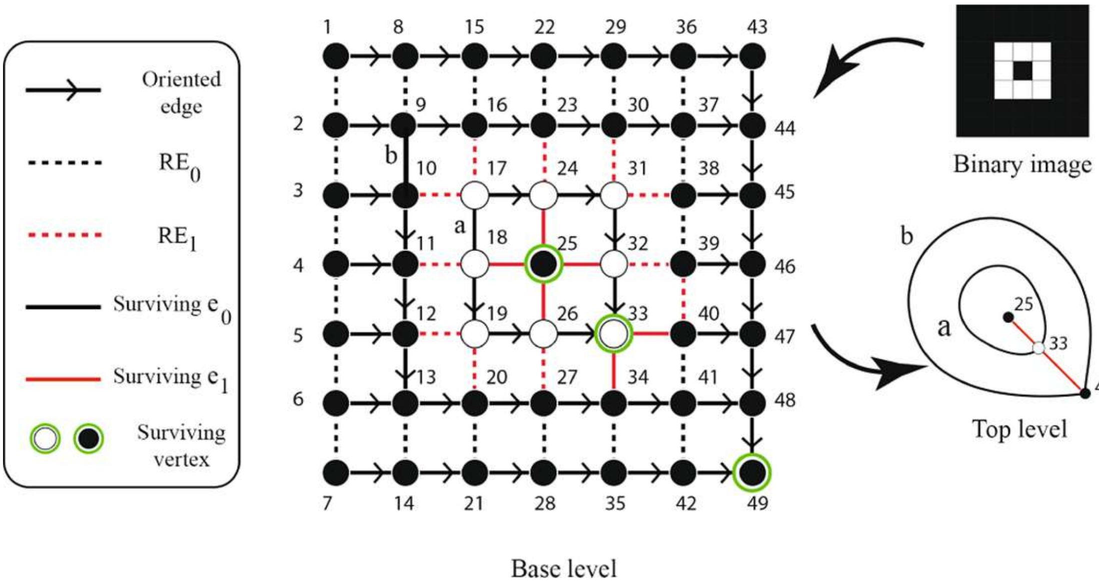


Fig. 3. Combinatorial map.

3.2 Redundant Edges

Connectivity is an essential property in the structure of a hierarchical graph pyramid. Nevertheless, there may be some edges the removal of which does not harm the connectivity. We define such edges as *redundant edges*.

Definition 5 (Redundant-Edge (RE)). *In an empty face, the non-oriented edge incident to the vertex with lowest Idx is redundant iff:*

- *The empty face is bounded by only non-oriented edges with the same contrast value.*
- *The empty face is bounded by non-oriented edges with the same contrast value and oriented edges.*

Based on the RE definition, an empty *self-loop* is redundant. In addition, in an empty face of degree 2 (*double edge*), one of the edges is redundant. Figure 4 illustrates an empty face of different degrees where in each empty face the redundant edge is indicated by RE.

Proposition 3. *The upper bound of the number of redundant edges (REs) is equal to half of the edges of the grid at the base level.*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

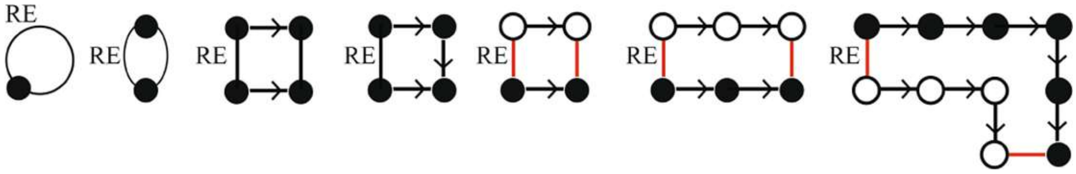


Fig. 4. Example of redundant edge (RE) in different empty faces.

Proof. In a grid M by N, the number of vertices is MN and the number of edges is $2MN - M - N$. To preserve the connectivity the smallest graph is a spanning tree of vertices with $MN - 1$ edges. Therefore, the maximum number of REs is:

$$Max|REs| = (2MN - M - N - (MN - 1)) = MN - M - N + 1 \tag{5}$$

$$\lim_{M\to\infty\ N\to\infty} (Max|REs|/E) = (MN - M - N + 1)/(2MN - M - N) = 1/2 \tag{6}$$

As the result, by growing M and N, the maximum number of REs becomes maximally half of all the edges (E) at the base level. □

Proposition 4. *In every face of degree* n $(n > 1)$ *is bounded by only* e_0s*, one of the non-oriented* e_0s *is redundant.*

Proof. By contracting an edge, every face of degree $n > 2$ after $n-2$ consecutive contractions becomes a face of degree 2 which has a RE (Definition 5). □

Proposition 5. *In every face of degree* n $(n > 1)$ *is bounded by only* e_1s *and oriented* e_0s*, there is a redundant one-edge* (RE_1) *.*

Proof. Contracting all oriented e_0s results in a face of degree 2 containing two e_1s between the same endpoints. Hence, one of the e_1s is redundant. □

Since edges classify into E_0 and E_1, the REs are partitioned into *Redundant Zero-Edges* (RE_0) and *Redundant One-Edges* (RE_1) as well:

$$REs = RE_0s \ \dot{\cup} \ RE_1s \tag{7}$$

In Fig. 3, the RE_0s are shown by black dashed-lines and the RE_1s are shown by red dashed-lines. Furthermore, the RAG at the top of the pyramid shows the connections between three different connected components. Using the combinatorial map structure, the inclusion relation is preserved because it is represented by the loop **a** aroundthe vertex 25.

It should be noted that a neighboring graph at the base level may not have any redundant edge. Consider a 4-connected graph that its vertices form a checkerboard pattern. In such the case, all edges have contrast one that it means there is no zero-edge and thus no RE_0. Furthermore, based on the Proposition 5, no two one-edges connect the same vertices and thus there is no RE_1 as well.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

3.3 Removing Redundant Edges in Parallel

In order to remove the REs, a dependency between edges is considered. We define such dependency relation to detect a set of REs where by simultaneously removing, the combinatorial structure is not harmed. To this aim, first a set of dependent darts is defined as follows:

Definition 6 (Dependent Darts). *All darts of a σ-orbit sharing an endpoint are dependent darts.*

Next, by considering the corresponding edge of each dart, $e = (d, \alpha(d))$, the set of dependent darts results in the set of **dependent edges**. Consequently, two edges not sharing an endpoint are independent. In this manner, the only case of the dependency between REs occurs when the REs share an endpoint. In the grid at the base level the REs may be connected horizontally or vertically and thus are dependent. However, consider a horizontal edge in an odd row of the grid. This edge is independent to all other horizontal edges of other odd rows. Similarly, a vertical edge in an odd column is independent of all other vertical edges of other odd columns. Such independency exists between edges in even rows and even columns as well. Figure 5 shows the set of independent edges at the base. Therefore, all the edges in grid are classified to four independent classes of edges. Consequently, removing all edges belonging to each independence class (1, 2, 3 or 4) occurs simultaneously. This means, all the REs are removed in only four steps where each step has the complexity $\mathcal{O}(1)$. Therefore removing the redundant edges is performed in parallel.

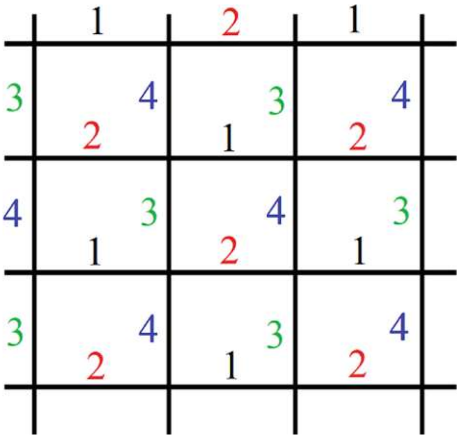


Fig. 5. The four independent classes of edges in the grid at the base.

4 Memory Consumption

The topological structure is well captured in the combinatorial map. A combinatorial pyramid is a hierarchy of successively reduced combinatorial maps [4]. The pyramid needs to store the combinatorial map of each level that results in

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

high memory consumption. To avoid such expensive memory requirement, we use a *canonical encoding* [17] where the memory consumption of the pyramid is equal to the size of the base level.

In the canonical encoding of the combinatorial pyramid, all the darts are stored in a single array that preserves the history of pyramid construction. The number of darts at the base level is equal to $2 \cdot (2MN - M - N)$ in a binary image M by N. Since nearly half of the edges at the base level (Proposition 3) are redundant (RE), their removal decreases the memory requirements.

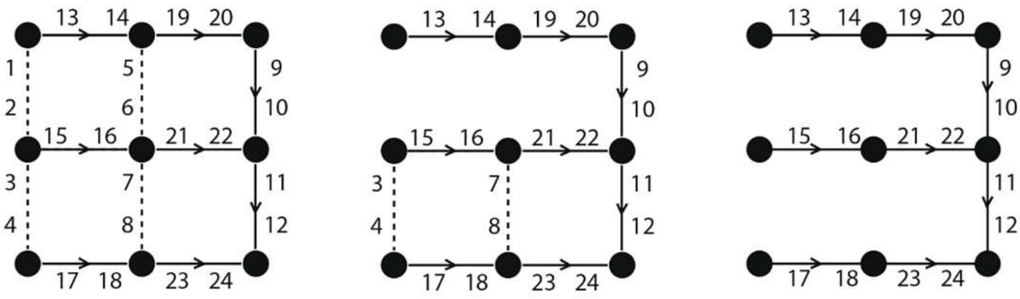
In addition, in the canonical pyramid, removing the darts performs in a sequential manner. In contrast, using the independent set of edges (Sect. 3.3) we are able to remove the independent set of corresponding darts in parallel. Therefore, in the canonical array such darts are removed in parallel. Fig. 6(a) illustrates the combinatorial map (CM) of a graph and Fig. 6(d) shows its canonical encoding. The REs are shown by dashed-lines. Four REs are corresponding to darts 1 to 8. The darts at the first row (1, 2, 5, 6) are removed in one step (Fig. 6(b, f)). Afterwards, darts at the second row (3, 4, 7, 8) are removed simultaneously (Fig. 6(c, g)). This results in, the smaller array of the canonical encoding shown in Fig. 6(h).

5 Comparisons and Results

To highlight the advantages of the proposed method, we compare the memory storage required with and without removing the REs. The comparison is done with the originally proposed canonical representation [17]. It was used by [1,3,6] for the implementation of topology preserving irregular image pyramids of gray scaled and RGB images. In addition, recently the canonical encoding was used in connected component labeling [2]. Since for our current research, the input images are restricted to binary images, it is easy to identify the connected components unlike the gray scale images. Considering the structure of the image, the number of REs that can be eliminated are significantly higher than that in a gray scale or RGB image.

In a combinatorial map, the involution α between the darts remain the same even after performing the contraction and/or removal operations. The α relations can be encoded into the even and odd numbering of darts for each edge. Thus all the modifications related to the contraction and the removal operation on the graphs are performed by modifying the σ-permutation. In the canonical representation, the minimum storage required to store and to modify the σ-permutation is equal to the number of darts i.e. twice the number of edges. By using the proposed method, we eliminate the edges that are structurally redundant and consequently reduce the storage space of darts and its permutation.

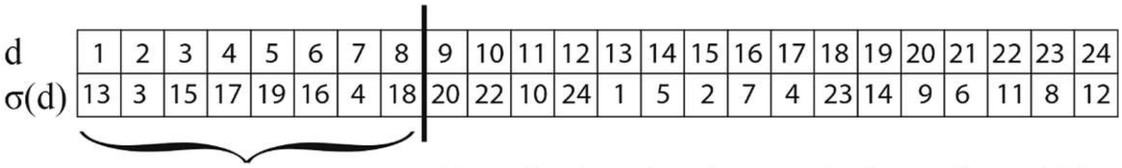
The algorithm was tested on several classes of images from the YACCLAB [8] dataset. Table 1 displays the outcome of the proposed method. The first column shows the name of the image class in the data set and an example from it, while the second column displays the 'size' of the image. The number of images ('#Images') from each class, on which the implementation was performed is



(a) CM of a graph (b) Removing REs of first row (c) Removing REs of second row

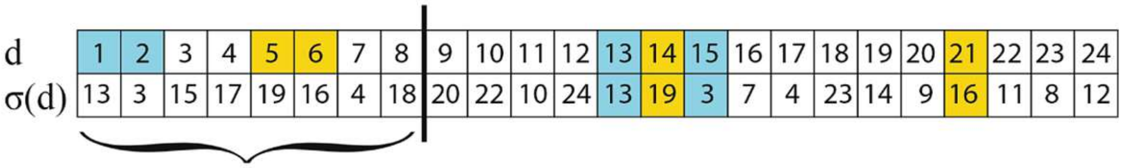
d	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
σ(d)	13	3	15	17	19	16	4	18	20	22	10	24	1	5	2	7	4	23	14	9	6	11	8	12

(d) Canonocal encoding of (a)



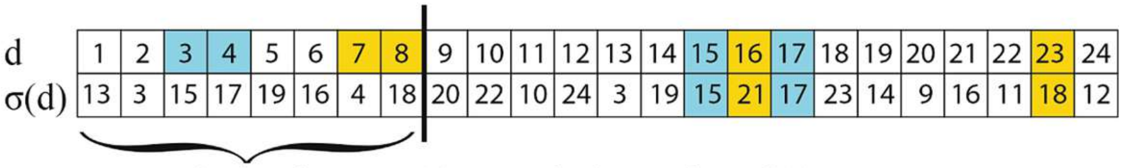
d	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
σ(d)	13	3	15	17	19	16	4	18	20	22	10	24	1	5	2	7	4	23	14	9	6	11	8	12

Redundant darts (e) Redundant darts in canonical encoding of (a)



d	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
σ(d)	13	3	15	17	19	16	4	18	20	22	10	24	13	19	3	7	4	23	14	9	16	11	8	12

Redundant darts (f) Canonical encoding of (b)



d	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
σ(d)	13	3	15	17	19	16	4	18	20	22	10	24	3	19	15	21	17	23	14	9	16	11	18	12

Redundant darts (g) Canonical encoding of (c)

d	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
σ(d)	20	22	10	24	3	19	15	21	17	23	14	9	16	11	18	12

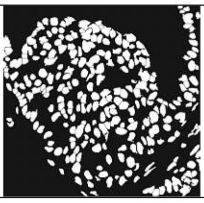
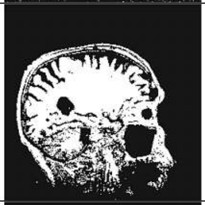
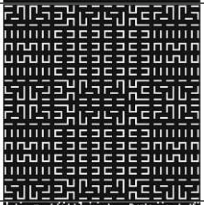
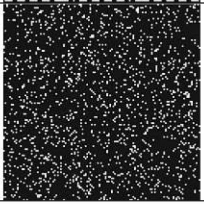
(h) CM after removing all redundant darts

Fig. 6. Memory usage in the canonical encoding.

displayed in the third column. The forth column gives the percentage of vertices that survive (‘$|V_s|/|V|$’). Since there is a significant variation in the size of the image, the number of REs are expressed in terms of percentage of the actual

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Table 1. Results over images of different categories from (YACCLAB [8]).

Database	Example	Size	#Images	$\lvert V_s\rvert/\lvert V\rvert$	$\lvert RE_{min}\rvert$	$\lvert RE_{\mu}\rvert$	std($\lvert RE\rvert$)
Mitochondria		768 × 1024	495	0.33%	49.01%	49.44%	0.0073
Medical		890 × 886	189	2.74%	41.18%	46.87%	0.0145
Finger-print		300 × 300	962	3.50%	42.50%	46.05%	0.0108
MRI		256 × 256	1170	2.72%	44.42%	46.49%	0.0114
3dpes		704 × 576	2400	0.07%	49.81%	49.84%	0.0019
Hilbert		127 × 127	512	2.43%	41.31%	45.25%	0.0108
Random		64 × 64	89	18.90%	23.18%	27.66%	0.0407

number of edges. The last three columns display the lowest ('$\lvert RE_{min}\rvert$'), and the average number of REs ('$\lvert RE_{\mu}\rvert$') along with the standard deviation ('std($\lvert RE\rvert$)') over all images from each dataset.

The redundancy in the random images is notably lower than that in the other class of images. This can be observed in the number of surviving vertices as well. This happens due to the fact that the number of isolated vertices (vertices

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

surrounded by e_1s only) are higher, making the connected component smaller in size. The worst case occurs in a checkerboard pattern where all the vertices are isolated making each region containing a single pixel. In such a case, none of the edges are redundant. In contrast, an image with only black (0) or only white (1) color will have 50% of the REs.

6 Conclusion and Future Works

The paper presents a new formalism to define redundant edges in the neighborhood graph of a 2D binary image. By proposing the new method for selecting the contraction kernels these redundant edges are efficiently detected and removed *before* the contraction operation. We prove that the amount of redundant edges may reach up to half of the edges at the base level with a grid like structure. The experiments show that most classes of images have 45%–49% of redundant edges (except for artificially generated random binary images). As a result, the memory consumption is reduced by 45%–49% while using combinatorial map as the data structure. Furthermore, all the redundant edges can be removed in parallel with a constant algorithmic complexity $\mathcal{O}(1)$. For the future work, we are going to develop the method for gray-scale images. Secondly, by using the combinatorial structure we will work on extending the redundant edges to higher dimensions (nD).

References

1. Banaeyan, M., Huber, H., Kropatsch, W.G., Barth, R.: A novel concept for smart camera image stitching. In: Čehovin, L., Mandeljc, R., Štruc, V. (eds.) Proceedings of the 21st Computer Vision Winter Workshop 2016, pp. 1–9 (2016)
2. Banaeyan, M., Kropatsch, W.G.: Pyramidal connected component labeling by irregular graph pyramid. In: 2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA), pp. 1–5 (2021). https://doi.org/10.1109/IPRIA53572.2021.9483533
3. Batavia, D., Gonzalez-Diaz, R., Kropatsch, W.G.: Image = structure + few colors. In: Torsello, A., Rossi, L., Pelillo, M., Biggio, B., Robles-Kelly, A. (eds.) S+SSPR 2021. LNCS, vol. 12644, pp. 365–375. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-73973-7_35
4. Brun, L., Kropatsch, W.: Introduction to combinatorial pyramids. In: Bertrand, G., Imiya, A., Klette, R. (eds.) Digital and Image Geometry. LNCS, vol. 2243, pp. 108–128. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45576-0_7
5. Brun, L., Kropatsch, W.G.: Hierarchical graph encodings. In: Lézoray, O., Grady, L. (eds.) Image Processing and Analysis with Graphs: Theory and Practice, pp. 305–349. CRC Press (2012)
6. Cerman, M., Janusch, I., Gonzalez-Diaz, R., Kropatsch, W.G.: Topology-based image segmentation using LBP pyramids. Mach. Vision Appl. **27**(8), 1161–1174 (2016). https://doi.org/10.1007/s00138-016-0795-1
7. Chen, L., Zheng, L., Xia, D., Cai, X., Sun, D., Liu, W.: Recognizing and analyzing private car commuters using big data of electronic registration identification of vehicles. IEEE Trans. Intell. Transp. Syst. pp. 1–15 (2022). https://doi.org/10.1109/TITS.2022.3142778

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

8. Grana, C., Bolelli, F., Baraldi, L., Vezzani, R.: YACCLAB - yet another connected components labeling benchmark. In: 2016 23rd International Conference on Pattern Recognition (ICPR), pp. 3109–3114. Springer (2016). https://doi.org/10.1109/ICPR.2016.7900112
9. Helman, J., Hesselink, L.: Visualizing vector field topology in fluid flows. IEEE Comput. Graph. Appl. **11**(3), 36–46 (1991). https://doi.org/10.1109/38.79452
10. Huang, W., et al.: A fast point cloud ground segmentation approach based on coarse-to-fine Markov random field. IEEE Trans. Intell. Transp. Syst. pp. 1–14 (2021). https://doi.org/10.1109/TITS.2021.3073151
11. Kropatsch, W.G.: Building irregular pyramids by dual graph contraction. IEE-Proc. Vision Image Signal Process. **142**(6), 366–374 (1995)
12. Kropatsch, W.G., Leonardis, A., Bischof, H.: Hierarchical, adaptive and robust methods for image understanding. Surv. Math. Ind. No. **9**, 1–47 (1999)
13. Kropatsch, W.G., Macho, H.: Finding the structure of connected components using dual irregular pyramids. In: Cinquième Colloque DGCI, pp. 147–158. LLAIC1, Université d'Auvergne (1995). ISBN 2-87663-040-0
14. Kropatsch, W.G., Reither, C., Willersinn, D., Wlaschitz, G.: The dual irregular pyramid. In: Chetverikov, D., Kropatsch, W.G. (eds.) CAIP 1993. LNCS, vol. 719, pp. 31–40. Springer, Heidelberg (1993). https://doi.org/10.1007/3-540-57233-3_4
15. Meer, P.: Stochastic image pyramids. Comput. Vision Graph. Image Process. **45**(3), 269–294 (1989)
16. Montanvert, A., Meer, P., Rosenfield, A.: Hierarchical image analysis using irregular tessellations. IEEE Trans. Pattern Anal. Mach. Intell. **13**(4), 307 (1991)
17. Torres, F., Kropatsch, W.G.: Canonical encoding of the combinatorial pyramid. In: Proceedings of the 19th Computer Vision Winter Workshop, pp. 118–125 (2014)
18. Wang, C., Wang, K., Liu, W.: Personalized recommendation via enhanced redundant eliminated network-based inference. In: 2019 First International Conference on Digital Data Processing (DDP), pp. 1–6 (2019). https://doi.org/10.1109/DDP.2019.00011
19. Wang, N., Haihong, E., Song, M., Wang, Y.: Construction method of domain knowledge graph based on big data-driven. In: 2019 5th International Conference on Information Management (ICIM), pp. 165–172 (2019). https://doi.org/10.1109/INFOMAN.2019.8714664
20. Zhang, D., Jiang, Y.: Design of urban intelligent traffic congestion situation monitoring system based on big data. In: 2020 International Conference on Intelligent Transportation, Big Data Smart City (ICITBS), pp. 12–15 (2020). https://doi.org/10.1109/ICITBS49701.2020.00011

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Paper B

Parallel $O(log(n))$ Computation of the Adjacency of Connected Components

Reprinted from [BK22b] with kind permission from Springer Nature.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Parallel $\mathcal{O}(log(n))$ Computation of the Adjacency of Connected Components

Majid Banaeyan( ) and Walter G. Kropatsch

TU Wien, Pattern Recognition and Image Processing Group 193/03, Vienna, Austria
`{majid,krw}@prip.tuwien.ac.at`

Abstract. Connected Component Labeling (CCL) is a fundamental task in pattern recognition and image processing algorithms. It groups the pixels into regions, such that adjacent pixels have the same label while pixels belonging to distinct regions have different labels. The common linear-time raster scan CCL techniques have a complexity of $\mathcal{O}(image - size)$ in a 2D binary image. To speed up the procedure of the CCL, the paper proposes a new irregular graph pyramid. To construct this pyramid, we use a new formalism [1] that introduces an order of the pixels in the base grid to detect the redundant edges through the hierarchical structure. These redundant edges, unlike the usual methods of constructing the irregular pyramid, are removed before contracting the edges. This not only simplifies the construction processes but may decrease memory consumption by approximately half. To perform the CCL task efficiently the proposed parallel algorithm reduces the complexity to $\mathcal{O}(log(n))$ where the n is the diameter of the largest connected component in the image. In addition, using an efficient combinatorial structure the topological properties of the connected components including adjacency of CCs, multi-boundaries and inclusions are preserved. Finally, the mathematical proofs provide fully parallel implementations and lead to efficient results in comparison with the state-of-the-art.

Keywords: Connected Component Labeling · Irregular graph pyramid · Parallel processing · Combinatorial map · Pattern recognition

1 Introduction

Connected Component Labeling (CCL) is used in analysing binary images as a basic task [16]. Given as input a binary image, its values distinguish between background (zero) or foreground (one) regions. After this, a region is **connected** if all pairs of pixels are connected by a chain of neighbors. They may be multiple regions with value zero and multiple regions with value one. CCL assigns a unique label to each different region. In general, the CCL algorithms divide into

---

Supported by Pattern Recognition Image Processing (PRIP) group, Vienna, Austria.

© Springer Nature Switzerland AG 2022

M. El Yacoubi et al. (Eds.): ICPRAI 2022, LNCS 13364, pp. 102–113, 2022.
https://doi.org/10.1007/978-3-031-09282-4_9

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

two main categories [11] based on label-propagation [10] or label-equivalence-resolving [12]. All of these approaches are linear and a pixel usually is visited in the raster-scan search. In other words, such algorithms may differ from one-scan or two-scan searching through the entire image, but all of them are in the order of image size, $\mathcal{O}(MN)$ in a $M \times N$-sized (2D) binary image. Recently, the algorithm proposed in [3] uses a pyramid structure for the CCL. However, because of the linear propagation of the labels, it is linear as well.

In contrast, In this study, the proposed Parallel Pyramidal Connected Component (//ACC[1]) method reduces the complexity impressively to the logarithmic order of the diameter of the largest connected component in the image. To this aim, we employ a new formalism in [1] to recognize the redundant edges in the pyramid. Removing these redundant edges *before* contracting the edges, not only is performed in parallel but may decrease memory consumption by half in comparison with efficient pyramids [17].

To construct the irregular pyramid, the **R**emove **t**hen **C**ontract (RtC) algorithm is proposed. The proposed algorithm speeds up the labeling task which makes it more efficient to be used in various application areas of machine learning and artificial intelligence such as document analysis and object recognition [15].

1.1 Motivations and Notations

Irregular pyramids are a stack of successively reduced graphs where each graph is constructed from the graph below by selecting a specific subset of vertices and edges. For generation of irregular pyramids, two basic operations on graphs are needed: edge contraction and edge removal. The former contracts an edge $e = (v, w)$, identifies v and w and removes the edge. All edges that were incident to the joined vertices will be incident to the resulting vertex after the operation. The latter removes an edge from the graph, without changing the number of vertices or affecting the incidence relationships of other edges. Note that in this study for preserving topology the self-loops are not contractable. In each level of the pyramid, the vertices and edges disappearing in level above are called *non-surviving* and those appearing in the upper level *surviving* ones.

Definition 1 (Contraction Kernel (CK). *A contraction kernel is a spanning tree of the connected component with the surviving vertex as its root.*

Each contraction kernel is a tree including one surviving vertex and the remaining non-surviving vertices of the CC.

A ***plane*** graph is a graph embedded in the plane such that its edges intersect only at their endpoints [18]. In plane graph there are connected spaces between edges and vertices and every such connected area of the plane is called a *face*. The *degree* of the face is the number of edges bounding the face. In addition a face bounded by a cycle is called an *empty face*. In a non-empty face traversing the boundary would require to visit vertices or edges twice.

---

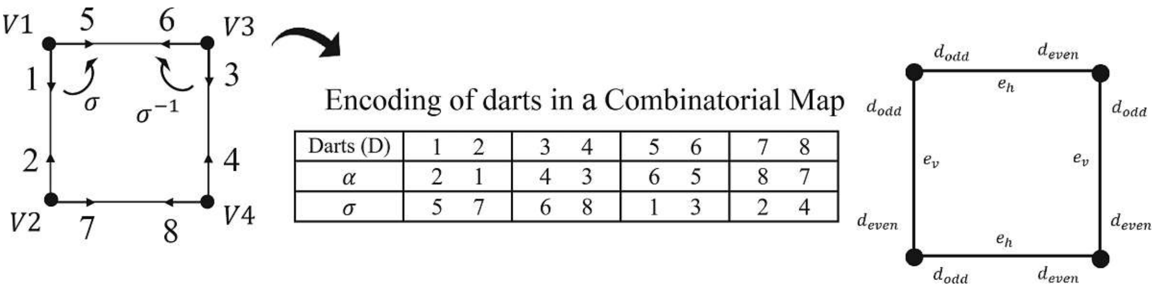
[1] It is pronounced pac where the // and A stand for parallel and pyramidal.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

There are different structures to build the irregular pyramid such as simple graphs [7], dual multi-graphs [13] and combinatorial maps (CM) [6]. The simple graph cannot distinguish some topological configurations (inclusions and multiple adjacency) [14]. The problem with dual graphs is that they cannot unambiguously represent a region enclosed in another one on a local level [7]. Therefore, in this paper the CM is used which not only resolves the mentioned problems but also can be extended to higher dimensions (nD).

A **combinatorial pyramid** is a hierarchy of successively reduced combinatorial maps [6]. A combinatorial map (CM) is similar to a graph but explicitly stores the orientation of edges around each vertex. To this aim, a permutation, σ, is defined encoding consecutive edges around a same vertex while turning counterclockwise. The clockwise orientation is denoted by σ^{-1}. In the CM each edge divides into two half-edges. Each half-edge is called a *dart* and the α is an *involution* providing a one-to-one mapping between consecutive darts forming the same edge such that $\alpha(\alpha(d)) = d$.

Figure 1 left, shows a set of 8 adjacent darts with their σ relations in a face of degree 4. In the middle, it shows the encoding of the darts. For instance, consider $e = (1, 2)$ where $\alpha(1) = 2$, $\alpha(2) = 1$, $\sigma(1) = 5$. In this paper, we assign an odd number to the left-side dart of a horizontal edge while assigning an even number to its right-side dart. Similarly, we assign an odd number to the up-side dart of a vertical edge while assigning an even number to its down-side dart. The d_{odd} indicates an odd dart and the d_{even} indicates an even dart (Fig. 1 right).



Encoding of darts in a Combinatorial Map

Darts (D)	1	2	3	4	5	6	7	8
α	2	1	4	3	6	5	8	7
σ	5	7	6	8	1	3	2	4

Fig. 1. Combinatorial map [1]

2 The RtC Algorithm for Pyramid Construction

In the original irregular pyramids [7] selected edges are first contracted. Edge contraction has the main advantage to preserve the connectivity. But it has a side effect to produce multiple edges and self-loops. Some of these edges are necessary to properly describe topological relations like inclusions and multiple connections between the same vertices. However, many of them are not necessary and hence called *redundant*. Redundant edges are removed through the simplification procedure after the contractions. However, in the proposed **R**emove **t**hen **C**ontract **(RtC)** algorithm, the redundant edges are removed *before* the contractions and in a parallel way. To this aim, the RtC introduces a new formalism to

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

define the redundant edges (Sect. 2.3). Since the RtC is used for the CCL task, the input is a binary image where the 4-connectivity between pixels is assumed because the 8-connectivity would not create a plane graph.

2.1 Edge Classification

Consider the neighborhood graph $G(V, E)$ of a binary image P where the vertices V correspond to the pixels P and the edges E connect two vertices if the corresponding pixels are 4-neighbors. Let the gray-value of vertex $g(v) = g(p)$ where $p \in P$ is a pixel in image corresponding to vertex v. Let $contrast(e)$ be an attribute of an edge $e(u, v)$ where $u, v \in V$ and $contrast(e) = |g(u) - g(v)|$. Since we are working with binary images, the pixels (and corresponding vertices can) have only two gray values 0 and 1. Similarly the edge contrast can have only two possible values 0 and 1. The edges in the neighborhood graph can be classified into the following two categories:

Definition 2 (Zero-edge) *An edge is a zero-edge iff the contrast between its two endpoints is zero. The zero-edge is denoted by* e_0.

Definition 3 (One-edge) *An edge is a one-edge iff the contrast between its two endpoints is one. The one-edge is denoted by* e_1.

The set of edges classified as zero-edge is denoted as E_0 and the set of edges classified as one-edge is denoted as E_1. The edge set $E = E_0 \cup E_1$.

A connected component consists of E_0 and E_1 connects different CCs together. Thus, the proposed algorithm for doing the labeling task, only considers E_0 as the candidates of the selection of the CK. Figure 2 shows a binary image with its corresponding neighborhood graph. Edges E_0 are black while E_1 are red.

2.2 Selecting the Contraction Kernel

The way a CK is selected has a main role in detecting the redundant edges in the neighborhood graph. To this purpose, a total order defined over the indices of vertices. Consider the binary image has M rows and N columns such that $(1, 1)$ is the coordinate of the pixel at the upper-left corner and (M, N) at the lower-right corner. An index $Idx(., .)$ of each vertex is defined:

$$Idx : [1, M] \times [1, N] \mapsto [1, M \cdot N] \subset \mathbb{N} \tag{1}$$

$$Idx(r, c) = (c - 1) \cdot M + r \tag{2}$$

where r and c are the row and column of the pixel(v), respectively. Since the set of integers is totally ordered each vertex has a unique index. The important property of such totally ordered set is that every subset has exactly one minimum and one maximum member (integer number). This property provides a unique orientation between non-surviving and surviving vertices. Consider a non-surviving vertex v. In order to find the surviving vertex, v_s, an incident

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

e_0 must be found in its neighborhood. Such a neighborhood $\mathcal{N}(v)$ is defined as follows [1]:

$$\mathcal{N}(v) = \{v\} \cup \{w \in V | e_0 = (v, w) \in E_0\} \tag{3}$$

if such neighborhood exists ($|\mathcal{N}(v)| > 1$) the surviving vertex is:

$$v_s = \text{argmax}\{Idx(v_s)|\ v_s \in \mathcal{N}(v), |\mathcal{N}(v)| > 1\} \tag{4}$$

Definition 4 (Orientation of a e_0). *A $e_0 = (v, w) \in E_0$ is oriented from v to w if w has the largest index among the neighbors, $Idx(w) = \max\{Idx(u)|u \in \mathcal{N}(v)\}$. All edges to the other neighbors remain non-oriented.*

Based on the definition above, a chain of oriented edges connects each non-surviving vertex to its corresponding survivor vertex. In Fig. 2 the oriented edges are represented by an arrow over each e_0. The surviving vertices (4, 12, 15), are presented by a green circle around each one.

In this paper, vertices surrounded by only $e_1 \in E_1$ are ***isolated vertices***. The isolated vertices will not be contracted through the construction process and they survive until the top of the pyramid. In the Fig. 2 the isolated vertices are 10 and 16 indicated by a blue circle.

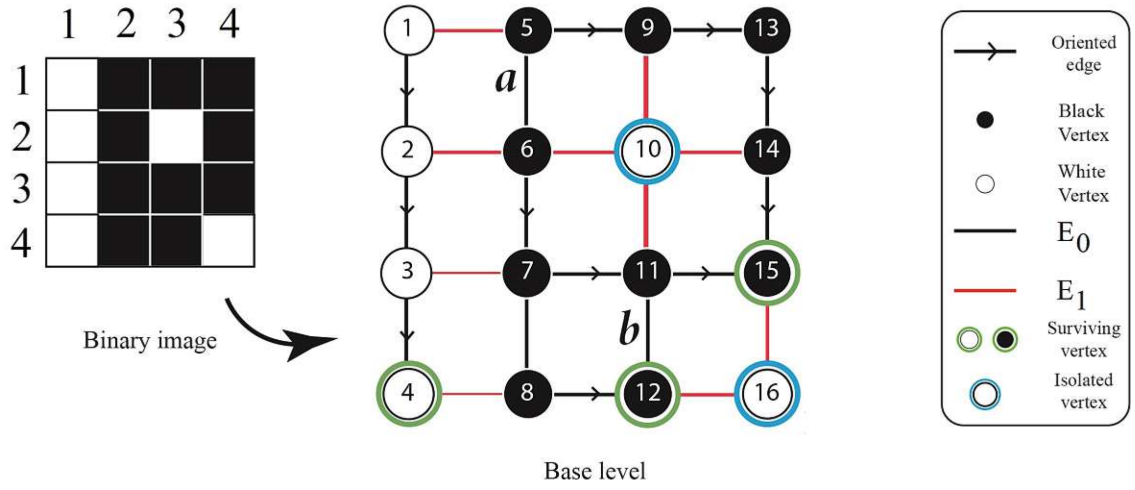


Fig. 2. The neighborhood graph of a 4 by 4 binary image.

2.3 Redundant Edges

In [1], redundant edges are investigated in details. To construct the irregular pyramid based on the RtC algorithm, first, the redundant edges are defined.

Definition 5 (Redundant-Edge (RE)). *In an empty face, the non-oriented edge incident to the vertex with lowest Idx is redundant iff:*

- *The empty face is bounded by only non-oriented edges with the same contrast value.*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

- *The empty face is bounded by non-oriented edges with the same contrast value and oriented edges.*

Proposition 1. *The upper bound of the maximum number of redundant edges (REs) is equal to half of the edges of the grid at base level.*

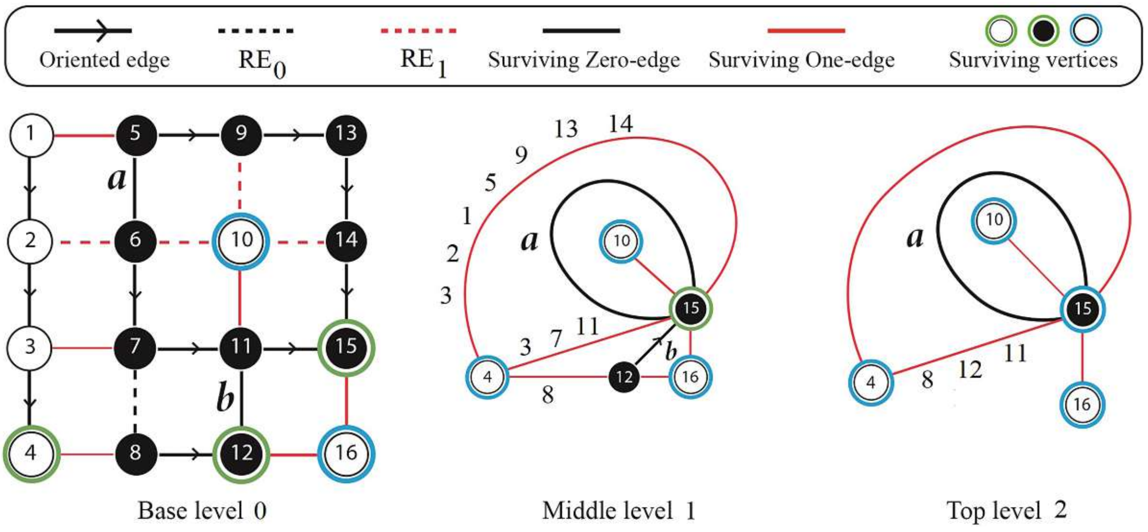
Proof. Can be found in [1]. ☐

Since edges classify into E_0 and E_1, the Redundant Edges (REs) are partitioned into *Redundant Zero-Edges* (RE_0) and *Redundant One-Edges* (RE_1) as well:

$$RE = RE_0 \ \dot{\cup} \ RE_1 \tag{5}$$

Removing RE and contracting the selected CKs at the base level, result in building the first level of the pyramid. To build the upper levels, the CKs are selected and then are contracted until there is no edge remaining for contraction. At this point, the pyramid reaches to its top level and the RtC algorithm is terminated.

In Fig. 3, different levels of the pyramid are shown. At the base level, the RE_0 is shown by a black dashed-line and the RE_1 are shown by red dashed-lines. Furthermore, the Region Adjacency Graph (RAG) of the middle and top level are illustrated. The RAG at top of the pyramid represents the connections between four different connected components. Using the combinatorial map structure, the **inclusion** relation is preserved as it is represented by the loop **a** around the vertex 10. Additionally, the structure preserves the **multiple boundaries** as it is shown by two different edges between vertices 4 and 15 with different paths of vertices (4-3-2-1-5-9-13-14-15 and 4-8-12-11-15) from the base level.



Base level 0 Middle level 1 Top level 2

Fig. 3. Binary irregular pyramid. (Color figure online)

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

2.4 Parallel Pyramidal Connected Component (//ACC)

The goal of connected component labeling is to assign a unique label to the vertices of a CC at the base level. Given a binary image as an input, first the corresponding pyramid is built by the RtC. At the top of the constructed pyramid, the RAG presents connected components (CCs) and the connectivity relations. Each CC is represented by one surviving vertex. The range of vertices between 1 to $M.N$ is kept. For each vertex a label as a new attribute is initialized. A surviving vertex at the top uses its index Idx as its unique label. To propagate down, each non-surviving vertex below the top level checks its parent and fills the label with the label of the parent. By reaching to the base level all the vertices receive their labels and the labeling task is finished. Since the CCL task is performed using the pyramid structure and in parallel, we call it **P**arallel **P**yramidal **C**onnected **C**omponent (//**ACC**).

3 Parallel Complexity

In this section the parallel complexity of the proposed //ACC algorithm is investigated. Whenever we talk about complexity, it is always assumed parallel complexity. The size of the binary input image is $M \times N$. Therefore, the indices of the vertices and the neighborhood relations of the edges are known. Note that such indexing is available *before* constructing the pyramid and in off-line processes. The edge classification and selection of the CKs are both performed locally over a vertex and its neighborhoods and therefore in parallel.

To remove the redundant edges (RE), a dependency between edges is considered. We define such dependency relation to detect a set of redundant edges where by simultaneously removing, the combinatorial structure is not harmed. Therefore, first a set of dependent darts is defined as follows:

Definition 6 (Dependent Darts). *All darts of a σ-orbit sharing an endpoint are dependent darts.*

Afterwards, by using the corresponding edge of each dart, $e = (d, \alpha(d))$, the set of dependent darts leads to the set of **dependent edges**. As a consequence, two edges not sharing an endpoint are independent. In this way, the only case of the dependency between RE occurs when the RE share an endpoint.

In the grid at the base level the RE may be horizontally or vertically connected and therefore are not independent. However, consider a horizontal edge in an odd row of the grid. This edge is independent to all other horizontal edges of other odd rows. Similarly, a vertical edge in an odd column is independent to all other vertical edges of other odd columns. Such independency occurs between edges in even rows and even columns as well. Figure 4.a , represents the set of independent edges at the base. Thus, all the edges in grid are classified into four independent set of edges.

As a result, removing all edge belong to each independent set (1, 2, 3 or 4), occurs simultaneously. This means, all the RE are removed in only four steps

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

where each step has the complexity $\mathcal{O}(1)$. Therefore removing the redundant edges is performed in parallel.

The disjoint sub-trees of a CK do not share any edges and therefore their contractions are performed in parallel. The challenging task is how to contract edges inside a tree of a CK in parallel? To this aim, we introduce two methods, one only for the base and the other for the remaining levels of the pyramid.

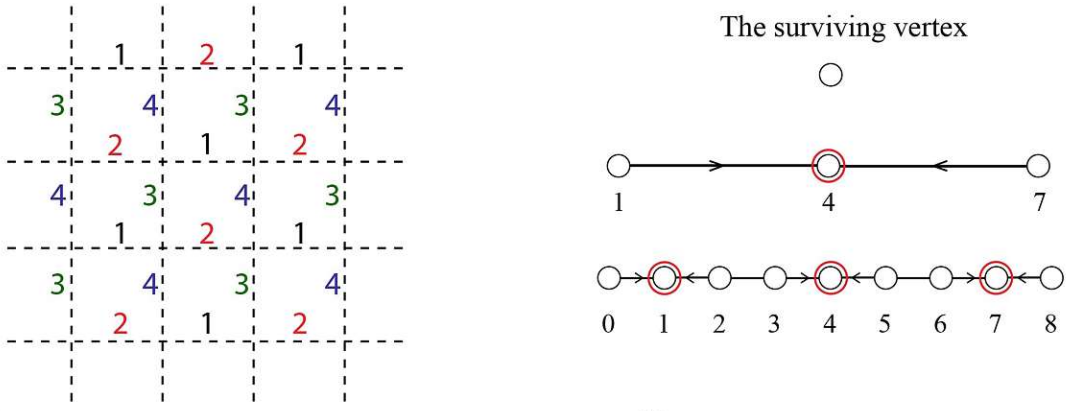
Contractions at the Base Level: Note that the diameter of a CK is the length of the largest path in the CK.

Proposition 2. *The complexity of contracting a CK has a logarithmic bound as follow:*

$$log_2(\delta(CK)) \leq complexity \quad of \quad contracting \quad a \quad CK \leq log_3(\delta(CK)) \tag{6}$$

where the $\delta(CK)$ is the diameter of the CK of the largest connected component in the image.

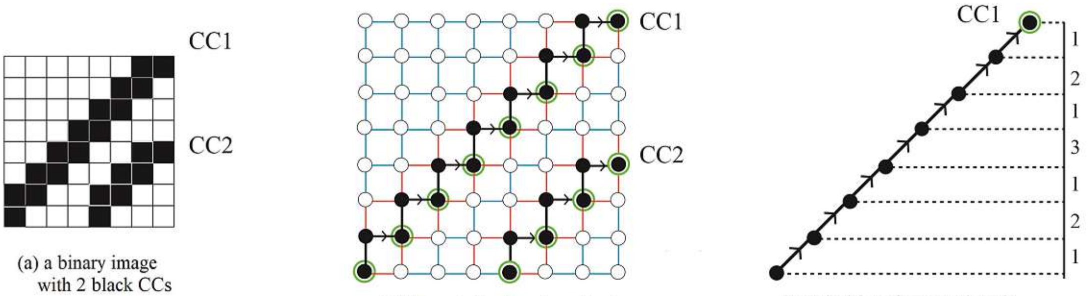
Proof. Based on (4), the maximum diameter of an oriented sub-tree graph corresponding to a $M \times N$ image is equal to $M + N - 1$. We consider a line sequence of edges with its length equal to this diameter. Next, the numbers from 0 to $M + N - 2$ are assigned to the vertices of the line sequence. By choosing the survivor vertices at $3n+1$ ($n \in \{0, 1, 2, [(M+N-2)/3]\}$), adjacent non-survivors ($3n$ and $3n + 2$) are contracted to this survivor (Fig. 4.b). Since each survivor belongs to a CK with the diameter at most 2, a line sequence consists of $3k$ vertices where $k \in \{1, 2, 3, ...\}$, needs only $log_3(3k)$ steps to select the survivors. The worst case occurs when the length of the line sequence is 4 and therefore, two steps ($log_2(4)$) are required for selecting the survivors. □



(a) The four independent set of edges (b) Contractions at the base level

Fig. 4. Independent edge sets [1], and contractions at the base

Contractions at Upper Levels: Based on (4), all remaining non-oriented E_0 are vertical edges at the base level (Fig. 5.b). The E_0 of two different CCs are disjointed and therefore they are independent. The E_0 of a CC may have at the worst case the $N-1$ vertical non-oriented edges in the neighborhood graph of the M by N binary image (CC1 in Fig. 5.b). These non-oriented edges receives their orientations at the next level ($L=1$) through the procedure of selecting the CKs and create a line of oriented edges. Such the line sequence of oriented E_0 are contracted in $\mathcal{O}(log_2(N-1))$ as visually is encoded in Fig. 5.c.



(a) a binary image with 2 black CCs

(b) The neighborhood graph of CC1 and CC2 at the base (L=0)

(c) Priorities of contractions in the line sequence of oriented E_0 (L=1)

Fig. 5. Priorities of contractions at level 1

4 Comparisons and Results

Simulations use MATLAB software and execute over CPU with AMD Ryzen 7 2700X, 3.7 GHz. The YACCLAB [9] benchmark was used for evaluating the proposed algorithm. The algorithm is executed over 89 random, 128 MRI and 128 finger-print images from this benchmark. Table 1 shows the results.

Table 1. Results over images of different categories from (YACCLAB[9]).

Database type	Random images	MRI images	Finger-print images
Size of the image	128 × 128	256 × 256	300 × 300
Redundant edges (average)	27.66%	46.49%	46.05%
Redundant edges (worst case)	23.18%	44.42%	42.50%
Number of connected components	2192	691	543
Execution time (ms) (in average)	0.098	1.643	2.317
Execution time (ms) (worst case)	0.127	2.973	3.518

The average percentage of the redundant edges (RE) over each category is represented. For example in the finger-print images, about 45% of the edges are redundant while they are all removed in parallel. Moreover, the number of connected components and the average time in each category are shown. The category of Random Images consist of only small objects. It means the diameter of a CK of the largest connected component of these small objects is negligible

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek, WIEN Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

in compare to the diameter of the image. Therefore the complexity is near to the $O(1)$. Essentially, the worst case occurs when the size of an object is as large as the whole image. In such the case, the complexity is equal to the logarithmic of the diameter of the image.

The inclusion relationship (hole) is one of the important topological information between connected components. The implementation of the proposed labeling //ACC, not only performs the labeling task, but also provides the number of inclusions between connected components. Furthermore, the simulations represent the adjacency and multi-adjacency of CCs. Such valuable topological information are missing in usual CCL algorithms. Figure 6 shows the CCL over a binary mitochondria image. The corresponding graph of the base level and categories of the edges are illustrated. The image consists of 9 connected component where the inclusion number is 7. In addition, the number of different edges for the mitochondria image are compared. The experimental results show approximately half of the edges in this image are RE.

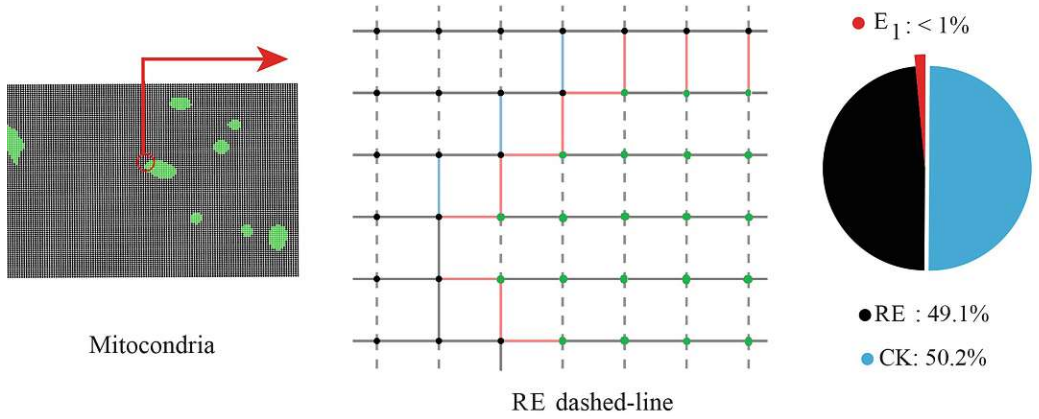


Fig. 6. A binary mitochondria image from [9]. Number of CCs is 9. The number of inclusions (holes) is 7. The RE are almost half of the edges.

Figure 7 shows the execution time of the //ACC algorithm over different image-sizes and compares it with the state-of-the-art methods from [5]. Although for small images the efficient algorithms in [5] are executed in higher speeds the //ACC with its logarithmic complexity reaches to the faster labeling results for big data, i.e., images larger than one million pixels.

Removing the RE not only speeds up the execution, but also decreases the memory consumption. The comparison is done with the originally proposed canonical represented [17] that also is used in [2,4,8]. In canonical representation, the minimum storage required to store the structure is equal to the number of darts i.e. twice the number of edges. By using the proposed RtC method, we eliminate the edges that are structurally redundant and consequently reduce the storage space of darts. Since the upper bound of the maximum number of RE is equal to half of the edges, the memory consumption of the proposed algorithm may decrease approximately by half.

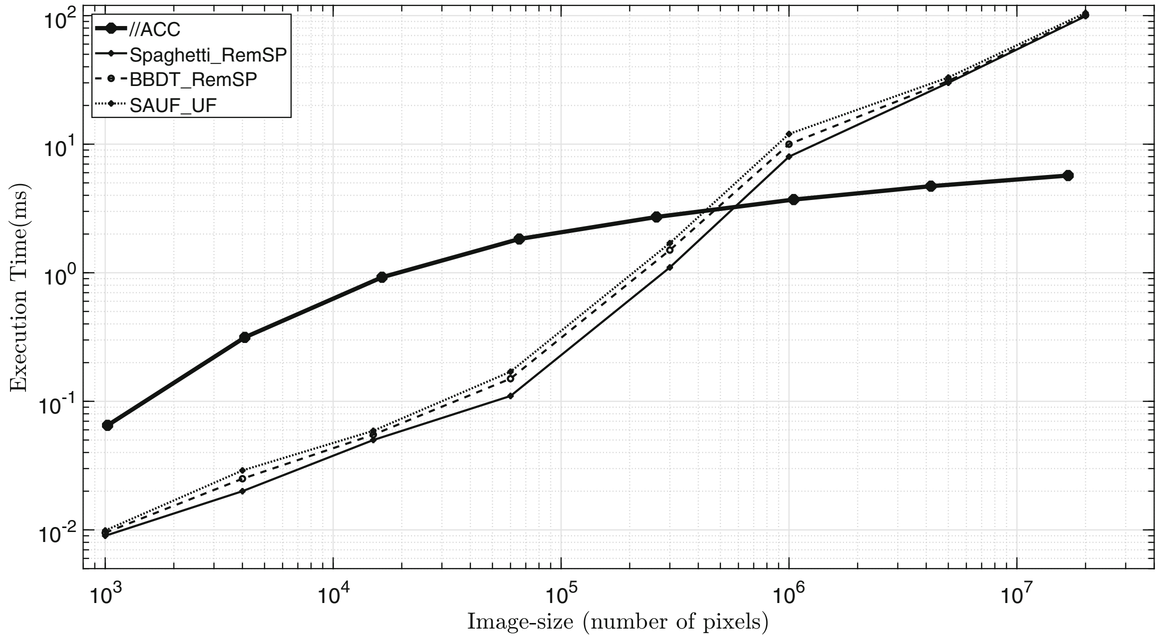


Fig. 7. Illustration of the execution time (ms) over different image-sizes

5 Conclusions

The paper presented a new approach to construct the irregular graph pyramids such that the connected component labeling can be performed in parallel and therefore faster. Unlike the usual construction of the irregular pyramids, in this paper, the redundant edges were removed in parallel *before* the contractions while they used to be removed after contractions and in a sequential order. The experimental results show that nearly half of the edges are removed as redundant edges that decreases the memory consumption to half of the combinatorial map of the base level of the pyramid. The logarithmic complexity of the algorithm speeds up the execution and suits it particularly for large images. In addition, the proposed method provides additional topological information such as inclusion and multi-boundaries. Moreover, what we proved it seems to be true for general graphs. Finally, using the combinatorial structure the proposed connected component labeling method can be extended to higher dimensions (nD) and to multi-label segmented images.

References

1. Banaeyan, M., Batavia, D., Kropatsch, W.G.: Removing redundancies in binary images. In: International Conference on Intelligent Systems and Patterns Recognition (ISPR), Hammamet, Tunisia, 24–25 March. Springer, Heidelberg (2022). (in print)
2. Banaeyan, M., Huber, H., Kropatsch, W.G., Barth, R.: A novel concept for smart camera image stitching. In: Čehovin, L., Mandeljc, R., Štruc, V. (eds.) Proceedings of the 21st Computer Vision Winter Workshop 2016, pp. 1–9 (2016)

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

3. Banaeyan, M., Kropatsch, W.G.: Pyramidal connected component labeling by irregular graph pyramid. In: 2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA), pp. 1–5 (2021)
4. Batavia, D., Gonzalez-Diaz, R., Kropatsch, W.G.: Image = Structure + Few colors. In: Torsello, A., Rossi, L., Pelillo, M., Biggio, B., Robles-Kelly, A. (eds.) S+SSPR 2021. LNCS, vol. 12644, pp. 365–375. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-73973-7_35
5. Bolelli, F., Allegretti, S., Baraldi, L., Grana, C.: Spaghetti labeling: directed acyclic graphs for block-based connected components labeling. IEEE Trans. Image Process. **29**, 1999–2012 (2020)
6. Brun, L., Kropatsch, W.: Combinatorial pyramids. In: Proceedings of International Conference on Image Processing, vol. 2, pp. II-33. IEEE (2003)
7. Brun, L., Kropatsch, W.G.: Hierarchical graph encodings. In: Lézoray, O., Grady, L. (eds.) Image Processing and Analysis with Graphs: Theory and Practice, pp. 305–349. CRC Press (2012)
8. Cerman, M., Janusch, I., Gonzalez-Diaz, R., Kropatsch, W.G.: Topology-based image segmentation using LBP pyramids. Mach. Vis. Appl. **27**(8), 1161–1174 (2016). https://doi.org/10.1007/s00138-016-0795-1
9. Grana, C., Bolelli, F., Baraldi, L., Vezzani, R.: YACCLAB - yet another connected components labeling benchmark. In: 2016 23rd International Conference on Pattern Recognition (ICPR), pp. 3109–3114. Springer, Heidelberg (2016)
10. He, L., Chao, Y., Suzuki, K.: Two efficient label-equivalence-based connected-component labeling algorithms for 3D binary images. IEEE Trans. Image Process. **20**(8), 2122–2134 (2011)
11. He, L., Ren, X., Gao, Q., Zhao, X., Yao, B., Chao, Y.: The connected-component labeling problem: a review of state-of-the-art algorithms. Pattern Recogn. **70**, 25–43 (2017)
12. Hernandez-Belmonte, U.H., Ayala-Ramirez, V., Sanchez-Yanez, R.E.: A comparative review of two-pass connected component labeling algorithms. In: Batyrshin, I., Sidorov, G. (eds.) MICAI 2011. LNCS (LNAI), vol. 7095, pp. 452–462. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25330-0_40
13. Kropatsch, W.G.: Building irregular pyramids by dual graph contraction. IEE-Proc. Vis. Image Signal Process. **142**(No. 6), 366–374 (1995)
14. Kropatsch, W.G., Macho, H.: Finding the structure of connected components using dual irregular pyramids. In: Cinquième Colloque DGCI, pp. 147–158. LLAIC1, Université d'Auvergne, ISBN 2-87663-040-0 (1995)
15. Qin, H., El Yacoubi, M.A.: End-to-end generative adversarial network for palm-vein recognition. In: Lu, Y., Vincent, N., Yuen, P.C., Zheng, W.-S., Cheriet, F., Suen, C.Y. (eds.) ICPRAI 2020. LNCS, vol. 12068, pp. 714–724. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59830-3_62
16. Shapiro, L.G.: Connected component labeling and adjacency graph construction. Mach. Intell. Pattern Recogn. **19**, 1–30 (1996)
17. Torres, F., Kropatsch, W.G.: Canonical encoding of the combinatorial pyramid. In: Proceedings of the 19th Computer Vision Winter Workshop, pp. 118–125 (2014)
18. Trudeau, R.: Introduction to Graph Theory. Dover Books on Mathematics (1993)

TU WIEN Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Paper C
Fast Distance Transforms in Graphs and in Gmaps

Reprinted from [BCKH22] with kind permission from Springer Nature.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Fast Distance Transforms in Graphs and in Gmaps

Majid Banaeyan( ), Carmine Carratù, Walter G. Kropatsch, and Jiří Hladůvka

Pattern Recognition and Image Processing Group, TU Wien, Vienna, Austria
{majid,krw,jiri}@prip.tuwien.ac.at

Abstract. Distance Transform (DT) as a fundamental operation in pattern recognition computes how far inside a shape a point is located. In this paper, at first a novel method is proposed to compute the DT in a graph. By using the edge classification and a total order [1], the spanning forest of the foreground is created where distances are propagated through it. Second, in contrast to common linear DT methods, by exploiting the hierarchical structure of the irregular pyramid, the geodesic DT (GDT) is calculated with parallel logarithmic complexity. Third, we introduce the DT in the nD generalized map (n-Gmap) leading to a more precise and smoother DT. Forth, in the n-Gmap we define n different distances and the relation between these distances. Finally, we sketch how the newly introduced concepts can be used to simulate gas propagation in 2D sections of plant leaves.

Keywords: nD distance transform · Generalized maps · Irregular pyramids · Parallel processing · Logarithmic complexity · Geodesic distance transform (GDT)

1 Introduction

The distance transform [5] computes for every pixel/voxel of an image/object how far it is from the closest obstacle, boundary, or background. While any valid metric may be involved in the computation of distance transforms, in topological data structures like graph, combinatorial maps [12], or generalized maps (n-Gmap) [7] often the shortest path between the obstacle/boundary and a given point is used. In this study, we first investigate the distance transform (DT) in graphs and then extend it to generalized map. We define different distances for every dimension (1D, 2D,..., nD) in the n-Gmap. This would be useful in many applications. In particular, in study of gas exchange through airspace of a leaf, computing the distances from stomata is very crucial to understand the different diffusion processes needed for photosynthesis.

Supported by the Vienna Science and Technology Fund (WWTF), project LS19-013, https://waters-gateway.boku.ac.at/.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
A. Krzyzak et al. (Eds.): S+SSPR 2022, LNCS 13813, pp. 193–202, 2022.
https://doi.org/10.1007/978-3-031-23028-8_20

Computing the DT and propagating the distances is an iterated local operation [8,15]. While local processes (e.g., convolution and mathematical morphology) are important in early vision, they are not suitable for higher level vision, such as symbolic manipulation and feature extraction where both local and global information is needed [9]. Therefore we exploit the advantage of the hierarchical structure of the pyramid [10] that encodes both local and global information similar to the human visual system [14].

In the pyramid there are two directions of processes: bottom-up and top-down. In the bottom-up (fine to coarse) process the information of the input data (e.g. intensity, color, texture) is transformed into global information. In the top-down (coarse to fine) process the global information such as the shape and the size of objects are refined into the base level of the pyramid. Therefore, the main idea of using hierarchical structure in computing the DT is to investigate the connectivity of a connected component in the local and general view within the pyramid. We will show that the connectivity can be checked in parallel logarithmic complexity instead of the linear raster scan commonly utilized in the state-of-the-art algorithms [8].

1.1 Notations and Definitions

An image P can be represented using a 4-adjacent neighborhood graph $G = (V, E)$ where V corresponds to pixels of P and E relates neighboring pixels. 8-Adjacency could be used only if the image is well-formed [11], which is not satisfied in general cases. The gray-value of a pixel $g(p)$ becomes an attribute of the corresponding vertex v, $g(v) = g(p)$ and the $contrast(e) = |g(u) - g(v)|$ becomes an attribute of an edge $e(u, v)$ where $u, v \in V$. In the neighborhood graph of the binary image, the edges have only two values: zero and one. We call them accordingly: **zero-edge** and **one-edge** [1]. Furthermore we denote the set of all zero-edges as E_0 and the set of all one-edges as E_1. In this way, the edges of the graph are partitioned into $E = E_0 \cup E_1$.

Irregular Pyramid. [10] is a stack of successively reduced smaller graphs where each graph is built from the graph below by selecting a specific subset of vertices and edges. In each level of the pyramid, the vertices and edges disappearing in level above are called *non-surviving* and those appearing in the upper level *surviving* ones.

Definition 1 (Contraction Kernel (CK)). *A CK is a tree consisting of a surviving vertex as its root and some non-surviving neighbors with the constraint that every non-survivor can be part of only one CK.*

Two basic operations are used to construct the pyramid: **edge contraction** and **edge removal**. In the edge contraction, an edge $e = (v, w)$ is contracted while its two endpoints, v and w, are identified and the edge is removed. The edges that were incident to the joined vertices will be incident to the resulting vertex after the operation. An arrow over an edge is commonly used to indicate the direction of contraction, i.e., from non-survivor to survivor (cf. Fig. 2). Contracting an edge has the enormous advantage of preserving the connectivity of the graph.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

During the edge removal, an edge is removed without changing the number of vertices or affecting the incidence relationships of other edges. Constraints are needed to make sure that edge removal does not disconnect the graph [6].

2 Distance Transform in a Graph

In a graph $G = (V, E)$ distances can be measured by the shortest length of paths. In this case the elements are the vertices V and neighbors $\mathcal{N}(v) = \{(v, w) \in E\}$ are related by edges. The distance between two vertices is the shortest path connecting the two vertices.

To compute the DT in a graph $G(V, E)$ with background $B \subset V$ and foreground $F \subset V$ vertices, the shortest distances of foreground vertices from the background should be computed. In this case the *seed* vertices $b \in B$ are initialized by $DT(b) = 0$. The foreground vertices $f \in F$ are initialized by $DT(f) = \infty$. Each one-edge $e = (b, f) \in E_1, b \in B, f \in F$ has two endpoints where $b \in B$ is a seed vertex with $DT(b) = 0$. The other vertex $f \in F$ belongs to the foreground and we initialize its distance by $DT(f) = 1$. The one-edges E_1 are frozen because they have no role in propagating the distances in the graph. Distances are propagated only through the E_0 edges of the foreground.

Using the total order on the foreground F proposed in [1,3], a spanning forest contains only edges E_0 spanning the foreground. The spanning forest is created in a single step with parallel constant complexity. Moreover, to propagate the distances we use the breath-first search (BFS) [4].

Proposition 1. *The parallel complexity of propagating DT is $\mathcal{O}(\delta(T))$ where $\delta(T)$ is the longest path in the spanning forest of the foreground.*

Proof. The complexity of propagating distances in a tree is $\mathcal{O}(|E|)$. Each connected component of the foreground is covered by a spanning tree which is processed independently [3]. Therefore, the longest path in the forest indicates the parallel complexity. □

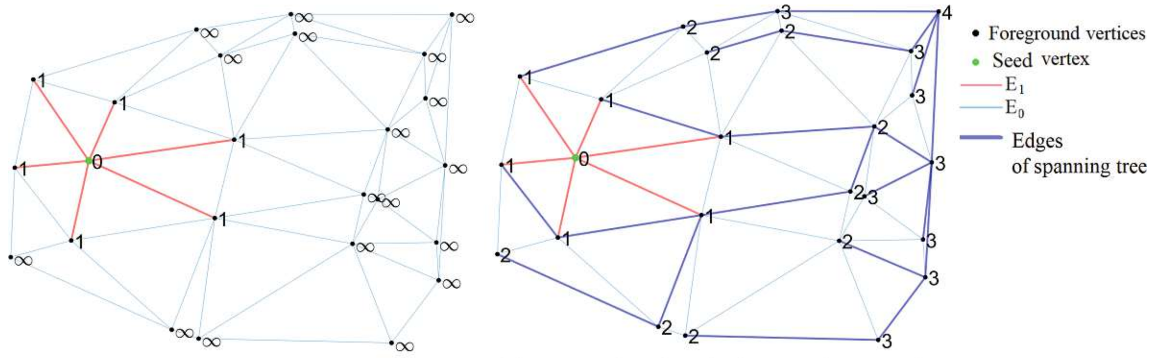
The propagation of the distances to the remaining vertices v of the foreground F follows:

$$D(v) = \min\{D(v), D(v_j) + 1 |\ v_j \in \mathcal{N}(v)\} \quad v \in F \tag{1}$$

where the foreground neighbors $\mathcal{N}(v)$ are defined by:

$$\mathcal{N}(v) = \{v \in F\} \cup \{w \in F | e_0 = (v, w) \in E_0\} \tag{2}$$

The distances are propagated until there is no vertex $v \in F$ with $DT(v) = \infty$ (see Fig. 1). Algorithm 1 shows the steps of computing the DT in a graph.



(a) initialization of DT

(b) propagation of DT in the spanning tree of F

Fig. 1. Computing the DT in a graph

Algorithm 1. Computing the DT in the Neighborhood Graph

```
**Input:** Neighborhood Graph: $G = (V, E) = (B \cup F, E_0 \cup E_1)$
Initialization: $DT(b) = 0 \ \forall b \in B, \quad DT(f) = \infty, \ \forall f \in F$
$DT(f) = 1 \ \forall (f, b) \in E_1, \quad f \in F, \quad b \in B$
**While** $\exists f \in F$ with $DT(f) = \infty$ **do**
    Propagate the distances by (1)
**end**
```

2.1 Geodesic Distance Transform

Geodesic DT (GDT) computes distances within the connected component of interest in a labeled image (or labeled neighborhood graph). The objects of interest are considered as the foreground objects and the remaining objects with different labels are considered as the background. A subset of points in the foreground are the seeds, $s \in S$, $S \subset F$, initialized by zero, $DT(s) = 0$. The aim is to compute the minimum distance of every point of the foreground to these seeds. The disjoint foreground objects keep the infinite distance if there is no seed in the connected component.

To compute the GDT we employ the irregular graph pyramid with logarithmic complexity. Each vertex receives a unique index and a total order is defined over the indices [1,3] that results in an efficient selection of contraction kernels (CKs). The CKs are only selected from E_0 edges which propagate the distances. The propagating distances are a set of power-of-two numbers. In Fig. 2 edges of CKs are shown by an arrow pointing towards the surviving vertex. The propagating distance i is shown by $\boxed{\text{i}}$ over an edge. By default all edges propagate distances by 1. Each surviving edge propagates the distance equal to 2^i into its adjacent unlabeled vertex. Next, to speed up the propagation of the distances with a power of two, the independent edges of a CK are identified by employing a logarithmic encoding. This logarithmic encoding indicates the priority of contractions through the construction of the pyramid. In Fig. 2a the numbers $1, 2, 3$ and $1', 2', 3'$ indicate the primary priorities that are different for each adjacent edge. The bottom-up construction of the pyramid (Fig. 2(a) to (d)) terminates when there is no edge remaining for the contraction. In top of

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

the pyramid all surviving vertices have their distance values. At this stage the distances are propagated from top to down where the vertices with $DT(v) = \infty$ receive their distance from their adjacent vertices and adding the distance of an edge (Fig. 2(d) to (g)).

In order to correctly compute the GDT, each surviving vertex counts the number of contractions from its receptive field while this is not needed in computing the DT.

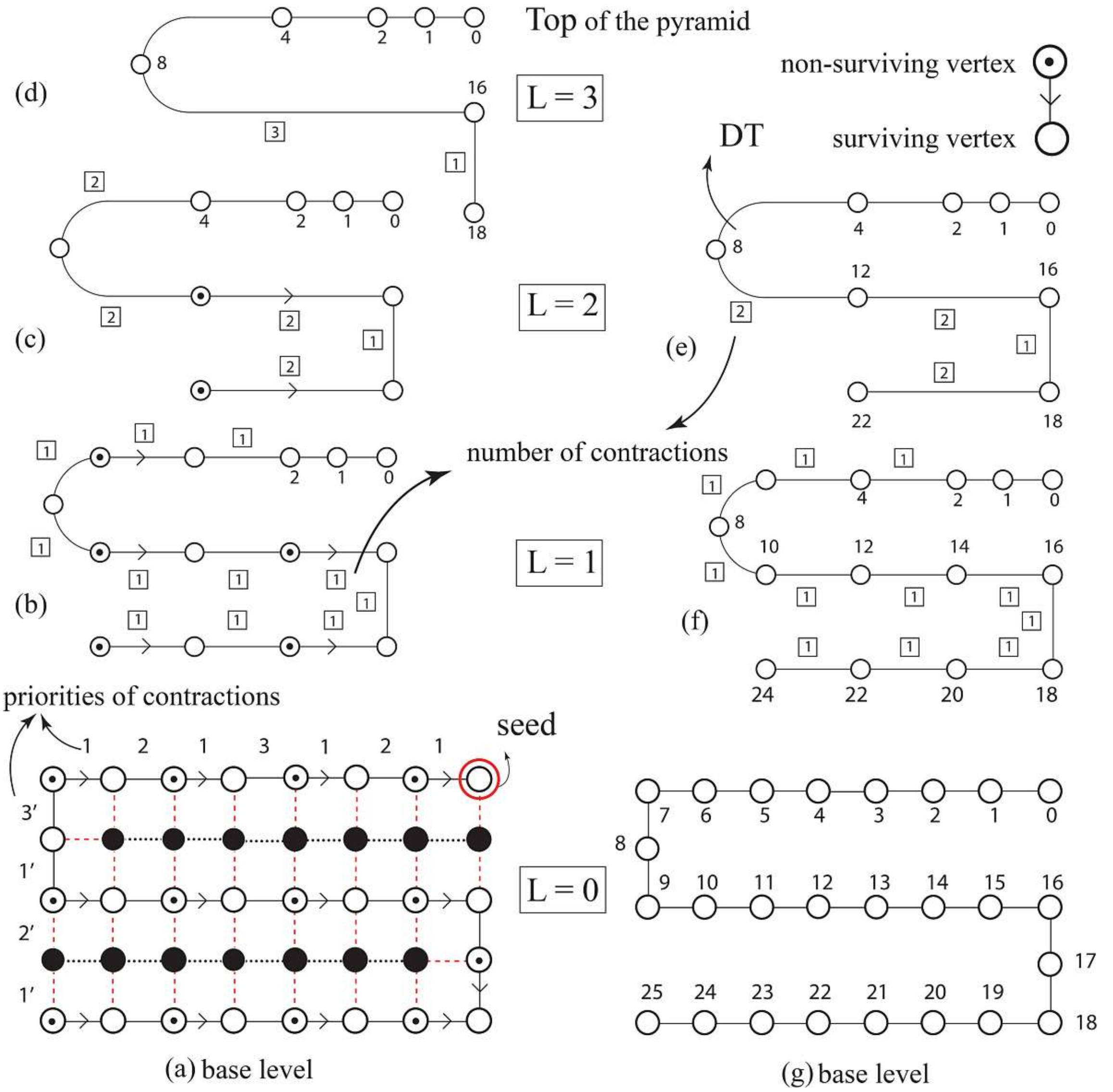


Fig. 2. Logarithmic GDT by irregular pyramid

Proposition 2. *Geodesic distance between two points in the higher dimension is always shorter or equal than in the lower dimension.*

Proof. Assume there is a distance between two points in the lower dimension that is shorter than distance between the same points in higher dimension. Since,

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

every point in lower dimension is included in higher dimension, the shorter distance in lower dimension exists in the higher dimension as well which is in contradiction with the assumption. □

3 Distance Transforms in n-Gmaps

An n-dimensional generalized map (n-Gmap) is a combinatorial data structure allowing to describe an n-dimensional orientable or non-orientable quasi-manifold with or without boundaries [12]. An n-Gmap is defined by a finite set of darts $\mathcal{D}$ on which act $n+1$ involutions[1] α_i, satisfying composition constraints of the following definition [7]:

Definition 2 (n-Gmap). *An n-dimensional generalized map, or n-Gmap, with* $0 \leq n$ *is an* $(n+2)$*-tuple* $G = (\mathcal{D}, \alpha_0, ..., \alpha_n)$ *where:*

1. $\mathcal{D}$ *is a finite set of darts,*
2. $\forall i \in \{0, ..., n\}$*:* α_i *is an involution on* $\mathcal{D}$
3. $\forall i \in \{0, ..., n-2\}$, $\forall j \in \{i+2, ..., n\}$*:* $\alpha_i \circ \alpha_j$ *is an involution.*

Let $(\mathcal{D}, \alpha_0, ..., \alpha_n)$ be an n-Gmap and let us consider its darts $d \in \mathcal{D}$ to be of a unit length. Similar to graphs, we first initialize the distance transform at any nonempty subset of *seed* darts $\mathcal{S} \subseteq \mathcal{D}$ as follows: $\delta(s) := 0 \; \forall s \in \mathcal{S}$ and $\delta(\bar{s}) := \infty$ $\forall \bar{s} \in \mathcal{D} \setminus \mathcal{S}$. Scenarios for the initialization (seeding) may include:

- single dart: $\mathcal{S} = \{d_0\}$,
- single i-cell: $\mathcal{S} = \{$all darts of the i-cell$\}$ (e.g., an edge), or
- any multi-combinations of the above, e.g., all edges (1-cells) connecting vertices of different labels resulting from segmentation or connected component labeling.

Similar to graphs, the distances are propagated from the seeds in the breath-first search. The difference to graphs, however, is that the propagation is more general and is driven along (some or all) *involutions* α_i rather than being restricted to the edges of the graph.

Figure 3b shows an example of a 2-Gmap – a 6×6 matrix of vertices (0-cells) of four labels A, B, C, and D where A and B have both two connected components. Edges $(d, \alpha_0(d), \alpha_2(d), \alpha_2(\alpha_0(d))$ connecting different labels[2] are initialized to 0 and distances are propagated following α_0, α_1, and α_2 involutions. Figure 3a illustrates the arrangement of darts around an implicit vertex (X).

The propagation of distances in Fig. 3b is performed equally in all dimensions, i.e., involving all involutions α_i. Excluding a fixed α_j, the propagation is constrained to manifolds of dimensions j. This makes the computation of the *geodesic* distance transforms on n-Gmaps viable.

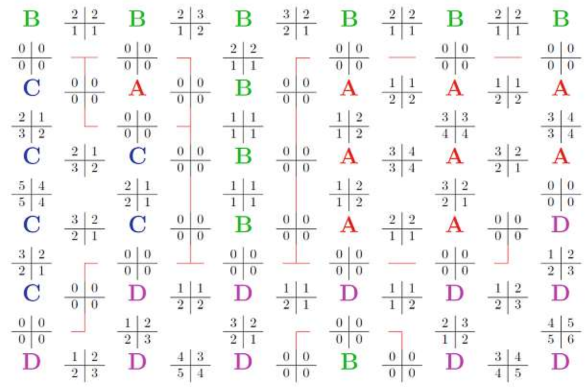
---

[1] Self-inverse permutations.

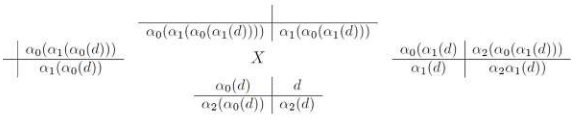
[2] red separators in Fig. 3(b).

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

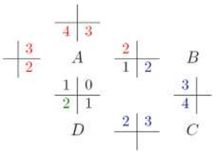
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub



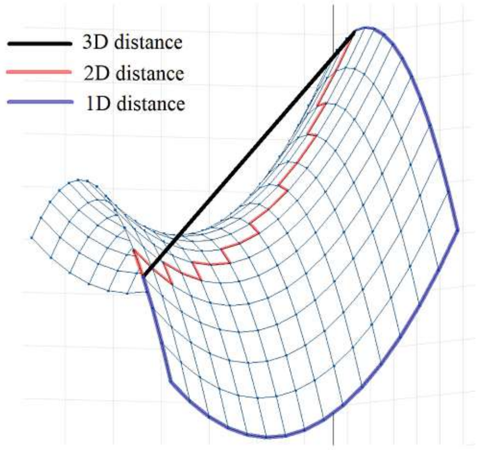
(b) Propagating of distances in the Gmap


$\alpha_0(\alpha_1(\alpha_0(d)))$
$\alpha_1(\alpha_0(d))$
$\alpha_0(\alpha_1(\alpha_0(\alpha_1(d))))$
$\alpha_1(\alpha_0(\alpha_1(d)))$
X
$\alpha_0(d)$
d
$\alpha_2(\alpha_0(d))$
$\alpha_2(d)$
$\alpha_0(\alpha_1(d)$
$\alpha_2(\alpha_0(\alpha_1(d)))$
$\alpha_1(d)$
$\alpha_2\alpha_1(d))$

(a) arrangement of darts around implicit vertex, X



(c) DT on manifolds of certain dimensions



(d) GDT in different dimensions

Fig. 3. DT in a Gmap (Color figure online)

We illustrate the effect by a simple 2D example (see Fig. 3c) where we initialize a single dart by zero and propagate distances only by pairs of involutions:

1. $\langle\alpha_0, \alpha_1\rangle$ denotes the propagation[3] of the orbit $(\alpha_0^*, \alpha_1^*)^*(d_0)$ and identifies the (dual) 2-cell between A,B,C,D. α_2 does not propagate the distance.
2. $\langle\alpha_0, \alpha_2\rangle$ denotes the propagation of $(\alpha_0^*, \alpha_2^*)^*(d_0)$ and identifies the 1-cell consisting of the four darts between A and D. In this case α_1 does not propagate the distance.
3. $\langle\alpha_1, \alpha_2\rangle$ denotes the propagation of $(\alpha_1^*, \alpha_2^*)^*(d_0)$ and identifies the 0-cell (a point), the eight darts surrounding A. In this case α_0 does not propagate the distance.

Depending on the initialization and the choice of involutions, distances can thus be propagated along the *boundaries* of any i-cells, $i > 0$. For 3-Gmaps, in addition to 3-cells (volume elements), propagation of distances along their (2D) bounding surfaces or along (1D) curves bounding these surfaces becomes possible. Based on Proposition.2 the GDT in the higher dimension is shorter or equal than in lower dimension (Fig. 3d).

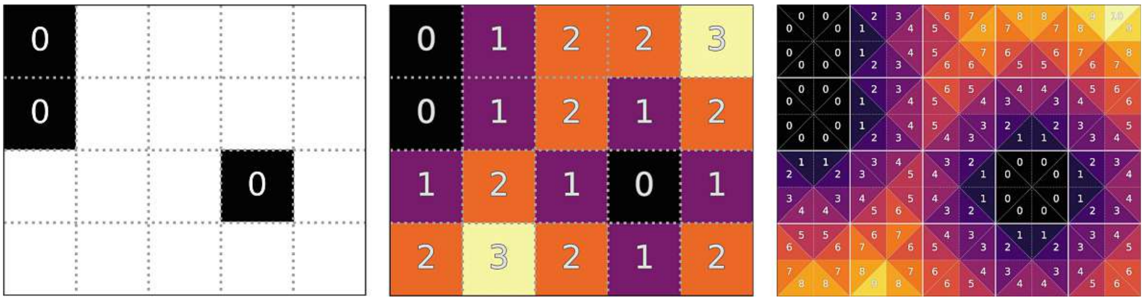
4 Results

As an example of the calculation of distance transforms on 2-Gmaps we refer to Fig. 4. The three black, zero-labeled pixels of the 4×5 image (Fig. 4a) are

---

[3] Blue distance values belong to the 2-cell, black distances to two types of cells (Fig. 3c).

used to seed the distance transform. Figure 4b represents the result of a graph-based distance transform where pixels correspond to vertices of the graph and its edges model the 4-connectivity of the image. The result of the 2-Gmap distance transform is displayed in Fig. 4c. Each pixel corresponds to eight darts which we choose to display by triangles colored by the minimum distance from the seeds. The axes-parallel and the diagonal lines between the triangles of one pixel correspond to α_0 and α_1, respectively. The axes-parallel *pixel-separating* lines correspond to α_2. The 3 seeds of Fig. 4a are represented by total of 24 black, zero-labeled triangles in Fig. 4c. It can be observed that in the 2-Gmaps the distances are propagated in a smoother and a more detailed way.

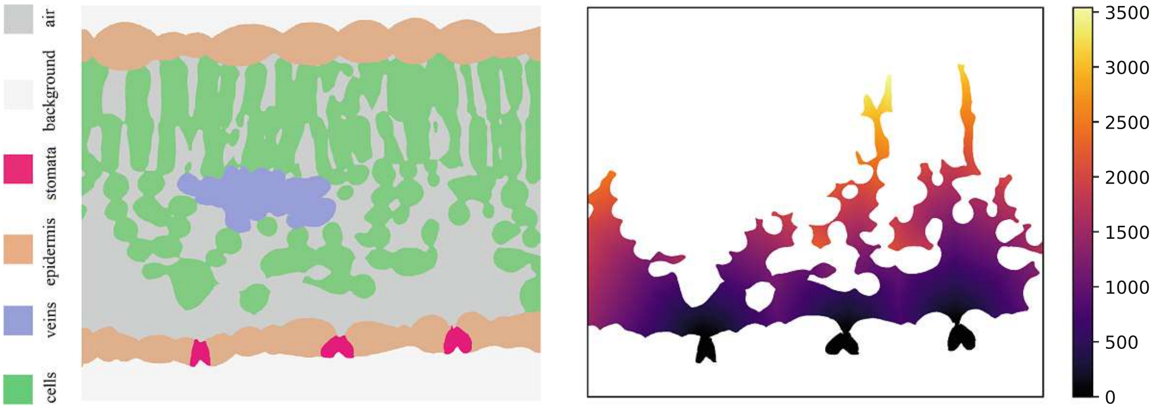


(a) mask with 3 seeds (b) DT (c) DT in Gmap

Fig. 4. Comparison of a graph-based (b) and Gmap-based (c) distance transforms of a binary image (a). Best viewed in color and magnified.

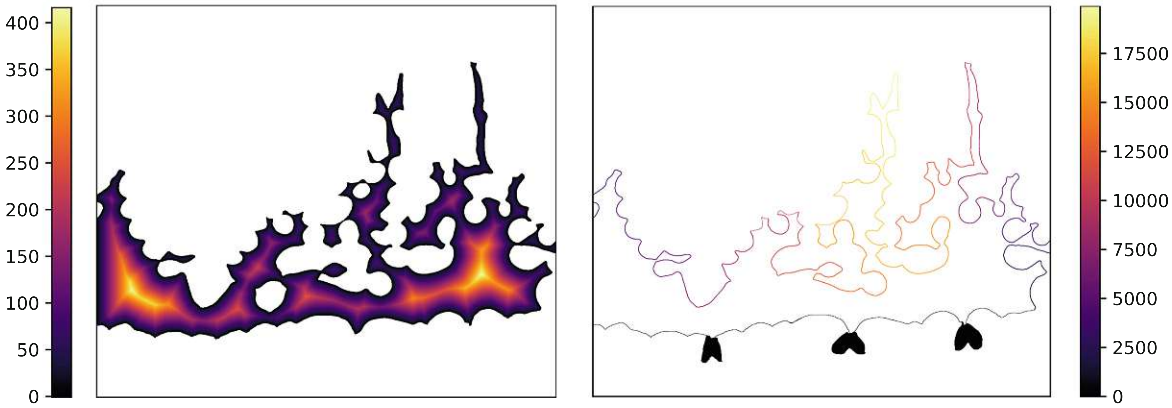
To exploit the advantage of the proposed method in a real application, several geodesic distance transforms (GDTs) are computed through a labeled 2D cross slice of a leaf scan (Fig. 5). The input image (Fig. 5a) has six different labels illustrating different regions inside the leaf. In this figure, the stomas act as gates to control the amount of CO_2 that is entering the leaf. The CO_2 propagates through the airspace to reach the cells and by combining with water and heat the photosynthesis takes place. To model various aspects of the photosynthesis, GDTs may aid in several ways. First, since we are interested in simulations of gas exchange in the leaf [13], we compute the GDT from the stomata through the airspace (Fig. 5b). This is intended to approximate how long it takes to reach the necessary CO_2 concentration. Second, bottlenecks of the airspace supposedly slow down the diffusion processes. We therefore compute the widths of the bottlenecks by the GDT inside the airspace seeded at ist boundary (Fig. 5c). Finally, the GDT from the stomata along the boundary of airspace is calculated (Fig. 5d). This is motivated by the observation that a longer boundary accommodates more cells to perform photosynthesis.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub



(a) Labeled cross-section of a plant leaf.

(b) GDT in air seeded at stomas.



(c) GDT in air seeded at its boundary.

(d) GDT in air boundary seeded at stomas.

Fig. 5. Computing GDT in a leaf.

It should be noted that the parallel logarithmic complexity of computing the GDT in the proposed method makes it useful for processing the big data. In our data-set each dimension of the 3D input image (leaf) is more than 2000 pixels. Therefore, fast computation of the DT with low complexity is required as shown in [2,3].

5 Conclusions

The paper presents a new algorithm to propagate distances in a graph which is based on a spanning forest of the foreground. The spanning forest is produced in parallel constant complexity and it reduces the linear search space to the length of the longest path in the spanning forest. By preserving the connectivity of connected components (CCs) and the topological information between CCs the proposed algorithm performs the connected component labeling (CCL) and the distance transform (DT) simultaneously. Using the hierarchical structure of the irregular pyramid the new method computes the geodesic distance transform (GDT) with parallel logarithmic complexity that makes it useful for processing of the big data.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

We additionally introduce distance transforms for the generalized combinatorial maps (n-Gmaps). We show how they naturally result in a smoother and a higher resolution distance fields. More importantly, however, we show how geodesic distance transforms can efficiently be performed just by omitting relevant involutions from the distance propagation. Finally, we demonstrate how computing GDTs in n-Gmaps may support modelling of the gas exchange in plant leaves.

References

1. Banaeyan, M., Batavia, D., Kropatsch, W.G.: Removing redundancies in binary images. In: International Conference on Intelligent Systems and Patterns Recognition (ISPR), Hammamet, Tunisia, 24–25 March 2022. pp. 221–233. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-08277-1_19
2. Banaeyan, M., Kropatsch, W.G.: Pyramidal connected component labeling by irregular graph pyramid. In: 2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA), pp. 1–5 (2021)
3. Banaeyan, M., Kropatsch, W.G.: Parallel $\mathcal{O}(log(n))$ computation of the adjacency of connected components. In: International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI), Paris, France, June 1–3, 2022. pp. 102–113. Springer, Cham (2022).https://doi.org/10.1007/978-3-031-09282-4_9
4. Beamer, S., Asanovic, K., Patterson, D.: Direction-optimizing breadth-first search. In: SC '12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. pp. 1–10 (2012). https://doi.org/10.1109/SC.2012.50
5. Borgefors, G.: Distance transformations in arbitrary dimensions. Comput. Vis., Graphics image Process. **27**(3), 321–345 (1984)
6. Brun, L., Kropatsch, W.G.: Hierarchical graph encodings. In: Lézoray, O., Grady, L. (eds.) Image Processing and Analysis with Graphs: Theory and Practice, pp. 305–349. CRC Press (2012)
7. Damiand, G., Lienhardt, P.: Combinatorial Maps: Efficient Data Structures for Computer Graphics and Image Processing. CRC Press (2014)
8. Fabbri, R., Costa, L.D.F., Torelli, J.C., Bruno, O.M.: 2D Euclidean distance transform algorithms: a comparative survey. ACM Comput. Surv. **40**(1), 1–44 (2008)
9. Haxhimusa, Y.: The Structurally Optimal Dual Graph Pyramid and Its Application in Image Partitioning. DISKI, Berlin (2007)
10. Kropatsch, W.G.: Building irregular pyramids by dual graph contraction. IEE-Proc. Visi. Image Signal Process. **142**(No. 6), 366–374 (1995)
11. Latecki, L., Eckhardt, U., Rosenfeld, A.: Well-composed sets. Comput. Image Underst. **61**(1), 70–83 (1995)
12. Lienhardt, P.: Topological models for boundary representation: a comparison with n-dimensional generalized maps. Comput. Aided Des. **23**(1), 59–82 (1991)
13. Momayyezi, M., et al.: Desiccation of the leaf mesophyll and its implications for CO_2 diffusion and light processing. Plant, Cell Enviro. **45**(5), 1362–1381 (2022)
14. Pizlo, Z., Stefanov, E.: Solving large problems with a small working memory. J. Probll. Solv. **6**(1), 5 (2013)
15. Rosenfeld, A., Pfaltz, J.L.: Sequential operations in digital picture processing. Assoc. Comput. Macch. **13**(4), 471–494 (1966)

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Paper D
Fast Labeled Spanning Tree in Binary Irregular Graph Pyramids

Reprinted from [BK22a] with kind permission from JENRS journal.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Received: 15 August, 2022, Accepted: 18 October, 2022, Online: 31 October, 2022
DOI: https://dx.doi.org/10.55708/js0110009

Fast Labeled Spanning Tree in Binary Irregular Graph Pyramids

Majid Banaeyan*,[1], **Walter G. Kropatsch**[1]
[1]Pattern Recognition and Image Processing Group, Institute of Visual Computing and Human-Centered Technology, TU Wien, Vienna, Austria
*Corresponding author: Majid Banaeyan, 1040 Wien, Favoritenstr. 9/5, E193-03, Vienna, Austria, +43 (1) 58801 - 18667 & majid@prip.tuwien.ac.at

ABSTRACT: Irregular Pyramids are powerful hierarchical structures in pattern recognition and image processing. They have high potential of parallel processing that makes them useful in processing of a huge amount of digital data generated every day. This paper presents a fast method for constructing an irregular pyramid over a binary image where the size of the images is more than 2000 in each of 2/3 dimensions. Selecting the contraction kernels (CKs) as the main task in constructing the pyramid is investigated. It is shown that the proposed fast labeled spanning tree (FLST) computes the equivalent contraction kernels (ECKs) in only two steps. To this purpose, first, edges of the corresponding neighborhood graph of the binary input image are classified. Second, by using a total order an efficient function is defined to select the CKs. By defining the redundant edges, further edge classification is performed to partition all the edges in each level of the pyramid. Finally, two important applications are presented : connected component labeling (CCL) and distance transform (DT) with lower parallel complexity $O(log(\delta))$ where the δ is the diameter of the largest connected component in the image.

KEYWORDS Spanning Tree, Irregular Pyramid, Total Order, Parallel Processing

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek, Your knowledge hub

1. Introduction

Pyramids are important structures in pattern recognition and image processing. They were invented [1] as ordered collection of images at multiple resolutions that are able to process high resolution data at lower resolution and propagating the local information into global and abstracted information at higher levels [2]. Pizlo [3] states that the pyramid is a general model for human problem solving where a massively parallel processing must be accomplished in order to recognizing a complex scene (like a busy street) in the blink of an eye [4, 5].

Motivated by a biological point of view, this paper introduces a fast method to construct the pyramidal structure of a given 2D binary image in a fully parallel scheme. Using the built pyramid, fundamental operations in analysing the binary images can be performed with lower complexity: Connected Component Labeling (CCL) and Distance Transform (DT). In particular, the current research is an extension of the previous work [6] that computes connected components (CCs) with the help of the pyramid. Propagating the labels in [6] is performed in linear time, hence the parallel complexity at the worst case is $O(\delta)$ where δ is the diameter of the largest CC in the image. In contrast, this paper mathematically proves that the parallel complexity is decreased to $O(log(\delta))$.

The paper is organized as follows. Sec. 1 gives a short overview of the theoretical background of image pyramids, graph pyramids and different graph representations. The classification of edges is defined in Sec. 2. Selecting the contraction kernels as the main step in constructing the irregular pyramid is completely described in Sec. 3. To this aim, the concept of *redundant* edges is covered by detail. The proposed fast labeled spanning tree (FLST) is defined in Sec. 4. Two main applications are presented in Sec. 5. The last section, provides a conclusion and considerations for future research.

1.1. Image Pyramids

Image Pyramids consist of a series of successively reduced images produced from a high resolution base image [2]. Generally, two types of the pyramids, namely regular and irregular pyramids exist. In regular pyramids [7] the resolution is decreased in regular steps and therefore the size of the pyramid is fixed. On the contrary, in irregular pyramids [8, 9] the size of the pyramid is not fixed and it is adapted to the image data. In addition, unlike the regular ones, the irregular pyramids are shift- and rotation-invariant which make them useful to use in a variety of tasks, in particular image segmentation [10, 11].

It should be noticed that the irregular image pyramid is interpreted as the **irregular graph pyramid** when its pixels and the neighborhood relations between adjacent pixels correspond to the vertices and the edges of the graph, respectively.

1.2. Irregular Graph Pyramids

Irregular pyramids are a stack of successively reduced graphs where each graph is constructed from the graph

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

below by selecting a specific subset of vertices and edges. For generation of irregular pyramids, two basic operations on graphs are needed: edge contraction and edge removal. The former contracts an edge connecting two vertices, and the two vertices are joined into one. All edges that were incident to the joined vertices will be incident to the resulting vertex after the operation. The latter removes an edge from the graph, without changing the number of vertices or affecting the incidence relationships of other edges.

In each level of the pyramid, the vertices/edges which disappear in a level above are called *non-surviving* vertices/edges. Those vertices/edges which appear in the upper level are called *surviving* vertices/edges. Consider $G = (V, E)$ as the neighborhood graph of an image P where V corresponds to the vertex set and E corresponds to the edge set. The vertex $v \in V$ associates with the pixels in image P and the edge $e \in E$ connects the corresponding adjacent vertices. Let the gray-value of vertex $g(v) = g(p)$ where $p \in P$ is a pixel in the image corresponding to vertex v. Consider $contrast(e)$ as an attribute of an edge $e(u, v)$ where $u, v \in V$ and $contrast(e) = |g(u) - g(v)|$ in the base level. Since we are working with binary images only, the vertices have either of the two values 0 and 1. Similarly the contrast of an edge is either 0 or 1.

Definition 1 (Contraction Kernel (CK))**.** *A CK is a tree consisting of a surviving vertex as its root and some non-surviving neighbors with the constraint that every non-survivor can be part of only one CK.*

An edge of a CK is denoted by the directed edge and points towards the survivor.
In this paper, the 4-connectivity between pixels of the input image is assumed. The reason is that the 8-connectivity would not be a plane graph [12]. A ***plane*** graph is a graph embedded in the plane such that its edges intersect only at their endpoints [13]. In a plane graph there are connected spaces between edges and vertices and every such connected area of the plane is called a *face*. The *degree* of the face is the number of edges bounding the face. In addition a face bounded by a cycle is called an *empty face*. In a non-empty face, traversing the boundary would require to visit vertices or edges twice [12].
An empty face consisting only one edge is called an empty *self-loop*. Consider an empty face of degree 2: it contains two edges that have the same endpoints. These parallel edges are called *multiple* edges. The multiple edges mean edges between the same endpoints, i.e. for example edges $e_{u_1,v_1} \neq e_{u_2,v_2} \neq e_{u_3,v_3}$ where $u_1 = u_2 = u_3$ and $v_1 = v_2 = v_3$.

1.3. Graph Representation

Graphs as a versatile representative tool are common in the representation of the irregular pyramid. There are different graph representations such as a *simple* graph, a *dual* graph and a *combinatorial* map.

A simple graph [14] $G = (V, E)$ consists of a set of vertices V and of edges E without *self-loops* and *multiple* edges between pairs of vertices. The relationships between different regions can be represented by the *region adjacency graph* (**RAG**). Although plane simple graphs are a common model for the RAG they cannot distinguish between different topological configurations, namely inclusion and multiple adjacency relationships (*multi-boundaries*) of regions [14].

A dual graph model encodes multiple boundaries between regions in a non-simple graph. The problem with dual graphs [9] is that they cannot unambiguously represent a region enclosed in another one on a local level [14]. Therefore, in this paper the combinatorial map (CM), as a planar embedding of a RAG, is used. It not only solves the mentioned problems but also provides an efficient structure to preserve topological relations between regions while it can be extended to higher dimensions (nD).

1.4. Combinatorial Pyramid

A combinatorial pyramid [15] is a hierarchy of successively reduced combinatorial maps. A combinatorial map (CM) is similar to a graph but explicitly stores the orientation of edges around each vertex. The 2D combinatorial map (G) is defined by a triple $G = (D, \alpha, \sigma)$ where the D is a finite set of darts [14]. A dart is defined as a half edge and it is the fundamental element in the CM's structure. The α is an *involution* on the set D and it provides a one-to-one mapping between consecutive darts forming the same edge such that $\alpha(\alpha(d)) = d$. The σ is a *permutation* on the set D and encodes consecutive darts around the same vertex while turning counterclockwise [16]. Note that the clockwise orientation is denoted by σ^{-1}.

Fig. 1a shows a set adjacent darts with their σ and α encoding. Note that the edge e between two vertices u and v is denoted by $e = (d, \alpha(d))$. The $u, v \in V$ and the $e \in E$ where the V and E are the set of vertices and edges of the graph $G = (V, E)$, respectively.
The removal and the contraction operations in the combinatorial pyramid is defined as follows:

Definition 2 (Removal operation)**.** *The removal operation removes one edge, $G \backslash \{e\}$, while it modifies the adjacent darts such that:*

$$\sigma(\sigma^{-1}(d)) = \sigma(d), \quad \sigma(\sigma^{-1}(\alpha(d))) = \sigma(\alpha(d)) \tag{1}$$

Definition 3 (Contraction operation)**.** *The contraction operation removes one edge, $G/\{e\}$, and collapses its two endpoints and modifies the adjacent darts such that:*

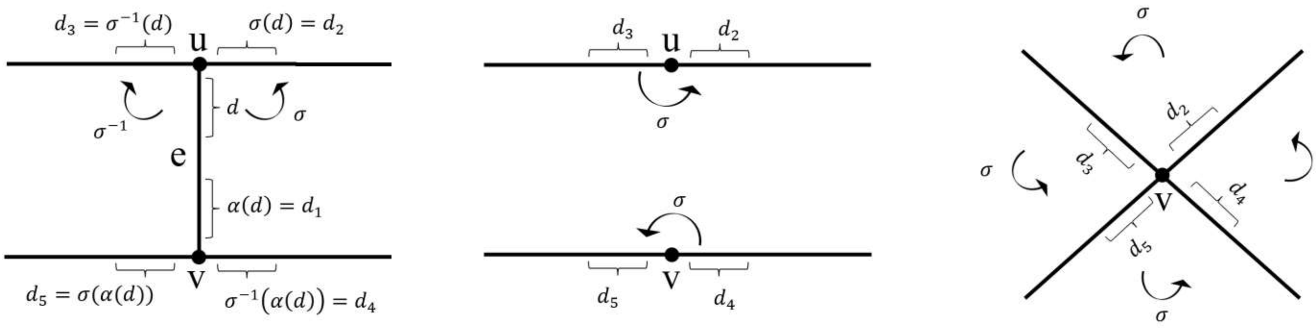
$$\sigma(\sigma^{-1}(d)) = \sigma(\alpha(d)), \quad \sigma(\sigma^{-1}(\alpha(d))) = \sigma(d) \tag{2}$$

Fig. 1b and Fig. 1c illustrate the removal and contraction operations in the combinatorial map. Note that the contraction operation does not disconnect the graph, and thus preserves connectivity [8].

2. Edge Classification in a Binary Image Graph

Let neighborhood graph $G = (V, E)$ be the undirected connected plane graph consisting of a finite set of vertices V and a finite set of edges E. In the neighborhood graph of the binary input image, each connected component (CC) consists of a set of vertices with the same gray value, 0 or 1. In the paper, black pixels (vertices) are shown by 0 while white pixels (vertices) are shown by 1. In this regard, we

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub



(a) An edge e with its incident darts in the CM.

(b) Removal operation, $G\backslash\{e\}$.

(c) Contraction operation, $G/\{e\}$.

Figure 1: Two main operations in irregular graph pyramids. (a) Before applying an operation. (b), (c) after applying the operations.

partition the edges of the neighborhood graph into two categories: edges connecting two vertices of the same CC, intra-CC and edges connecting vertices of different CCs, inter-CCs as follows:

Definition 4. *Intra-CC edge: an edge $e = (u, v) \in E$ within a CC is **intra-CC** iff $g(u) = g(v)$.*

Definition 5. *Inter-CC edge: an edge $e = (u, v) \in E$ between two CCs is **inter-CC** iff $g(u) \neq g(v)$.*

The contrast of an intra-CC edge is equal to zero, $c(intra\text{-}CC) = 0$. Therefore, we denote the intra-CC edge by $e_0 \in E_0$. The contrast of an inter-CCs edge is one, $c(inter\text{-}CCs) = 1$. Therefore, the inter-CCs edge is denoted by $e_1 \in E_1$. All edges in the neighborhood graph are partitioned into E_0 and E_1 edges:

$$E = E_0 \dot{\cup} E_1 \tag{3}$$

3. Selecting the CKs using a Total Order

Selecting the CKs plays the main role in constructing the irregular pyramid. The height of the built pyramid and the complexity of the construction depends on how the CKs are selected. In order to achieve an efficient and a unique selection of the CKs a **total order** is defined over the vertices [17]. Consider G as the neighborhood graph of an binary input image with M by N vertices. Let $(1, 1)$ be the coordinate of the vertex at the upper-left corner and (M, N) at the lower-right corner. Let r and c denote the row and the column in the grid structure of G, respectively. The vertices of G receive a unique index as follows:

$$Idx : [1, M] \times [1, N] \mapsto [1, M \cdot N] \subset \mathbb{N} \tag{4}$$
$$Idx(r, c) = (c - 1) \cdot M + r \tag{5}$$

We use the properties of the total order [18] in selecting the CKs. First, every two elements of a total ordered set (indices of vertices) are comparable. Second, each subset of the total ordered set (a set of vertices) has exactly one minimum and one maximum.

In the binary neighborhood graph G a CC consists of only intra-CC (E_0) edges. In constructing the irregular pyramid this CC is shown by only one single vertex at the top of the pyramid. Therefore, all the CKs are selected only from the intra-CC edges. From the vertex point of view, a vertex that is not incident to an intra-CC edge is an *isolated* vertex. This vertex is surrounded by only inter-CC edges.

Let v be a non-isolated vertex, i.e, it is the endpoint of at least one intra-CC edge. The *upper* neighborhood $\mathcal{N}$ is defined as follows :

$$\mathcal{N}(v) = \{w \in V | (v, w) \in E_0,\ Idx(w) > Idx(v)\} \tag{6}$$

The cardinality of the set $|\mathcal{N}(v)|$ indicates the number of intra-CC edges incident to v having greater vertex than v. Therefore, the cardinality of the non-isloated vertex is $|\mathcal{N}(v) \geq 1|$.

In order to determine the CKs in the graph $G = (V, E)$, the *selecting contraction kernel* **SCK(.)** function is defined as follows:

$$SCK : [1, M \cdot N] \mapsto [1, M \cdot N] \tag{7}$$
$$SCK(Idx(v)) = \max\{Idx(w) | \ w \in \mathcal{N}(v)\} \text{ if } |\mathcal{N}(v)| \geq 1 \tag{8}$$
$$SCK(Idx(v)) = Idx(v) \ \text{ if } \ |\mathcal{N}(v)| = 0 \tag{9}$$

The output of the SCK function partitions the vertices into two categories: surviving vertices and non-surviving vertices as follows:

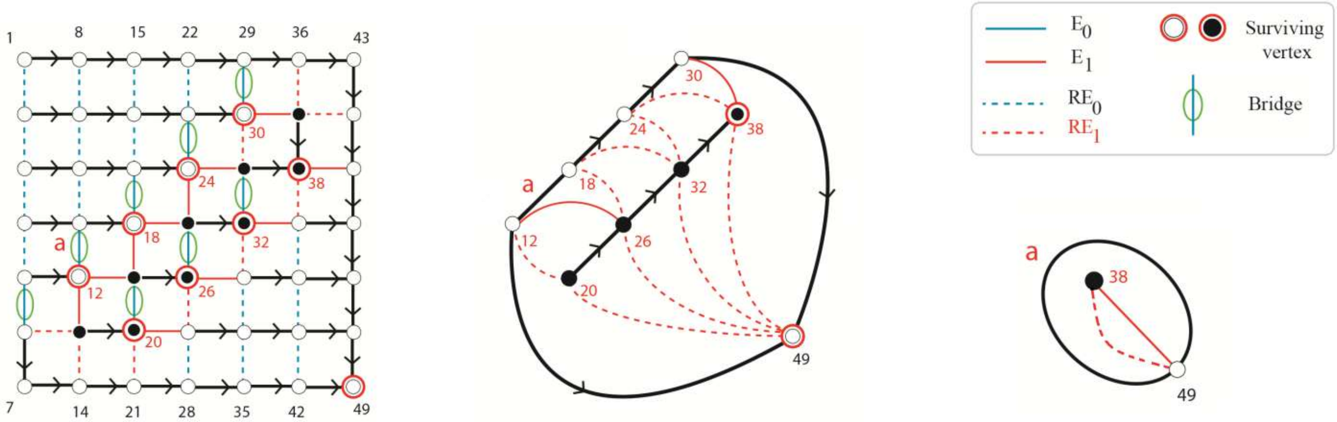
Definition 6. *[Surviving vertex] A vertex $v \in V$ in a binary neighborhood graph $G = (V, E)$, is a surviving vertex iff $SCK(Idx(v)) = Idx(v)$.*

Proposition 1. *An isolated vertex survives always.*

Proof. Assume v is an isolated vertex. Since there is no intra-CC edge incident to the isolated vertex, it leads to $|\mathcal{N}(v)| = 0$. Based on the (9), $SCK(Idx(v)) = Idx(v)$ and therefore employing the Def. 6 v is the surviving vertex. □

Definition 7 (Non-surviving vertex)**.** *A vertex $v \in V$ in a binary neighborhood graph $G = (V, E)$, is a non-surviving vertex iff $SCK(Idx(v)) \neq Idx(v)$.*

Proposition 2. *For a non-surviving vertex v at the base level, $|\mathcal{N}(v)| \in \{1, 2\}$.*



(a) First step of selecting the CKs at the base level.

(b) Second step of selecting the CKs.

(c) Top of the pyramid.

Figure 2: Two steps of selecting the CKs. The edge a shows the inclusion relationship between CCs.

Proof. Based on (5) a non-surviving vertex may be incident to maximum two vertices with greater indices (right or down vertices) at the base level. In addition, the non-surviving vertex must be incident to at least one vertex, namely its right or its down vertex. The former states $|\mathcal{N}(v)| = 2$ while the latter states $|\mathcal{N}(v)| = 1$. □

In a CK there is one surviving vertex (the root of the CK) while the remaining vertices are non-surviving vertices. Each non-surviving vertex connects to the surviving root by a unique monotonically Idx-increasing path of *oriented* edges. In a graph with n vertices there are $n!$ different total order[1]. Each selected total order has its own properties. Selecting an efficient total order effects on selecting the CKs where the number of CKs determines the height of the pyramid. Pyramids with logarithmic height reduce the parallel computational complexity of fundamental operations such as connected component labeling [19, 17] and distance transform [20]. Therefore, a proper selection of the total order must result in constructing the pyramid with logarithmic height. In Sec. 4.1 it is proved that the proposed total order leads to this logarithmic height.

In contrast to the common methods of constructing the pyramid [2, 8], using the proposed total order has the advantage that the vertices are partitioned in every level of the pyramid. In other words, the vertices are either the non-surviving or the surviving vertices. Next sections show how this partitioning reduces the number of steps in selecting the CKs into only two steps.

3.1. First Step of Selecting the CKs

Selecting the CKs at the base level of the pyramid is the first step of the selection. To this aim, the SCK function is performed over each vertex of the neighborhood graph of the base level. As the result, each CK has one surviving vertex and all the other vertices of the CK do not survive. In Fig. 2-a the surviving vertices at the base level are denoted by a red circle around each vertex while all the other vertices do not survive.

At the base level of the pyramid, all faces in the grid structure are bounded by four edges containing two horizontal and two vertical edges.

Proposition 3. *A horizontal intra-CC edge in a face of degree 4 at the base level of the pyramid always belongs to a CK.*

Proof. Assume a horizontal intra-CC edge $e = (u, v)$ at the base level does not belong to a CK. Let u and v locate at the left and the right side of the edge e, respectively. Since u is the endpoint of e, thus u is not an isolated vertex ($|\mathcal{N}(u)| > 1$) and based on (5), $Idx(u) < Idx(v)$. Due to Pro. 2 if $|\mathcal{N}(u)| = 1$ then v is the only vertex of $\mathcal{N}(u)$ that is incident to u and thus v is selected as the survivor. In case $|\mathcal{N}(u)| = 2$, there are two right and down vertices in the $\mathcal{N}(u)$ where based on (5) the right vertex is selected as surviving vertex. □

3.2. Redundant Edges

Graphs as a versatile representative tool may have many unnecessary (*redundant*) edges [17]. Through the construction of the pyramid, contracting edges results in a smaller induced graph at the upper level. The resulting graph may consist of empty self-loops or double-edges. At this point, the edge removal simplifies the graph and removes these redundant edges.

Definition 8 (Redundant Edges)**.** *In a hierarchical structure, those edges that are not needed to fully reconstruct the hierarchy are considered as* ***redundant edges****.*

Generally, the definition of the redundant edges depends on the applications and to what extend the reconstruction needs to be performed. For example, Banaeyan et al. [6, 17, 19] defined the concept of the redundant edges in a binary graph pyramid in order to do the connected component labeling task where the fully reconstruction is performed. They showed that the redundant edges can be detected (predicted) before performing the contraction of edges. In this paper, we use the same concept for defining the redundant intra-CC and redundant inter-CC edges.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

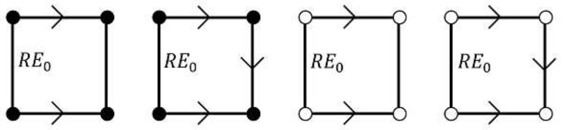
---

[1] All total orders are permutations of each other.

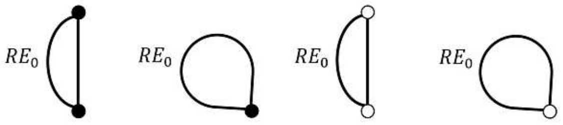
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Definition 9 (Redundant Intra-CC edge (RE_0))**.** *In an empty face consisting of only intra-CC edges, the non-oriented edge incident to the vertex with lowest Idx is a redundant intra-CC edge.*

The definition above states that a redundant intra-CC edge (RE_0) exists only in a face bounded by intra-CC edges. Fig. 3 illustrates the configuration of the redundant intra-CC edges.



(a) RE_0, before contracting the CKs.



(b) Corresponding RE_0, after contracting the CKs.

Figure 3: Configurations of the redundant intra-CC edges.

Definition 10 (Redundant Inter-CCs Edge (RE_1))**.** *In an empty face, an inter-CCs edge incident to the vertex with lowest Idx is redundant iff:*

- *The empty face consists of only two inter-CCs edges.*
- *The empty face is bounded by inter-CCs edges and oriented intra-CC edges.*

Fig. 4 illustrates all different configurations of the RE_1 edges before and after contracting the CKs.

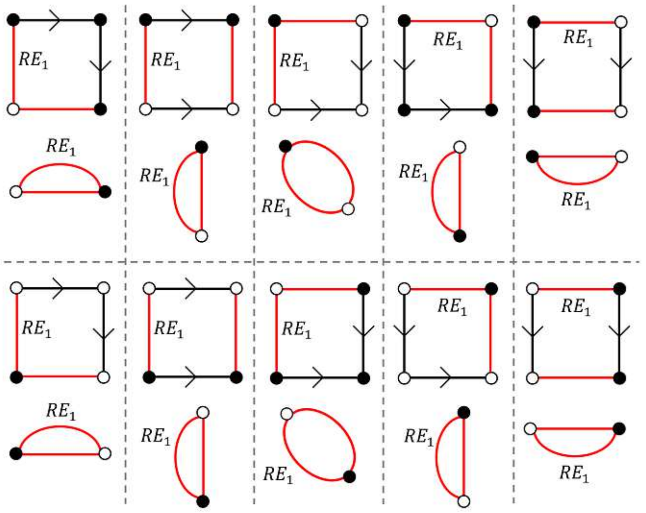


Figure 4: All different configurations of redundant inter-CCs edges

3.3. *Second Step of Selecting the CKs*

At the base level of the pyramid there are three types of the intra-CC edges:

1. The oriented edges that belong to the CKs.
2. The non-oriented redundant edges, RE_0, were defined in Def. 9.
3. The remaining non-oriented intra-CC edges are defined as the *bridges*.

Definition 11 (Bridge)**.** *A **bridge** is a non-oriented intra-CC edge that bridges the gap between two contraction kernels of a connected component.*

Note that the bridge is the edge of the equivalent contraction kernel (ECK) that is contracted after the two CKs are contracted.

Proposition 4. *A bridge in a face of degree 4 at the base level of the pyramid is the vertical edge.*

Proof. In the face of degree 4, there are two horizontal and two vertical edges. Assume the non-oriented bridge is the horizontal edge. However, due to Pro. 3 every horizontal intra-CC edge is oriented and therefore it cannot be a non-oriented intra-CC edge. □

Proposition 5. *A face of degree 4 at the base level of the binary pyramid does not have more than one bridge.*

Proof. Assume a face of degree 4 contains two bridges. Since the bridges are vertical intra-CC edges, the oriented intra-CC edge must connect two different CCs which is in contradiction with the definition of the oriented edge (see Fig. 5-b). □

Proposition 6. *Two bridges at the base level of the binary pyramid are not incident to a same vertex.*

Proof. Assume that two bridges are incident to the same vertex. Therefore, the horizontal common edge between their two corresponding faces must be the oriented intra-CC edge and the inter-CCs edge at the same time (see Fig. 5-c) , contradiction. □

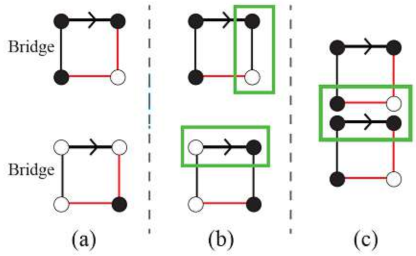
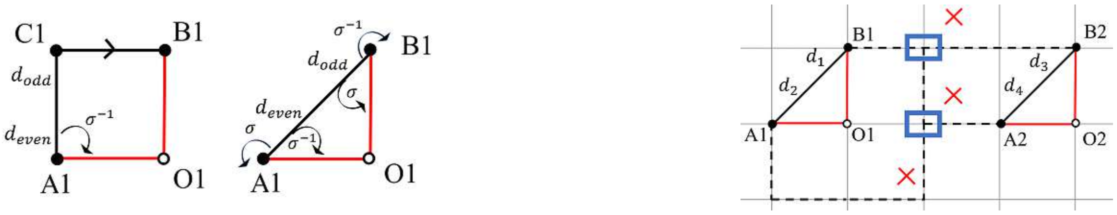


Figure 5: (a) The configurations of a bridge at the base level. (b) A face does not have two bridges. (c) Bridges are not incident to the same vertex

In order to select the CKs at the second step, the SCK function is performed over the bridges.

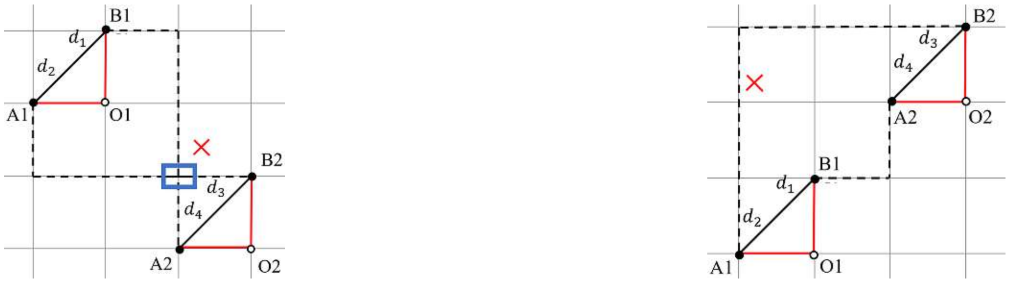
Definition 12 (inclusion edge)**.** *An inclusion edge is a non-empty self-loop inter-CC edge that preserves the topological inclusion relationship between two different CCs.*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN



(a) The configurations of a bridge before and after the edge contraction.

(b) Contradiction to planarity of the graph G.



(c) Contradiction to planarity of the graph G.

(d) Contradiction to (5).

Figure 6: All redundant intra-CC edges, RE_0, are at the base level of the pyramid.

Note that if the inclusion relationships exists between two different CCs, the inclusion edge is one of the bridges that will be detected after the contractions of oriented edges. In Fig. 2 the inclusion edge is shown by a non-empty self-loop that is denoted by the letter **a** with the red color.

Proposition 7. *All the redundant intra-CC edges are detected at the base level of the binary pyramid.*

Proof. The redundant intra-CC edges occur in a face bounded by only intra-CC edges. At an upper level, the remaining intra-CC edges are the bridges at the base level. However, since each bridge has the σ-related to a inter-CCs edge (Fig. 6a), therefore, there is no empty face containing only intra-CC edges at upper levels of the pyramid. Note that in the simple graph G, two bridges cannot be the σ-related of each other because this contradicts to planarity of the graph (Fig. 6b and Fig. 6c) or it contradicts to (5) that is shown in Fig. 6d. □

The Proposition .7 states that there is no redundant intra-CC edge at an upper level of the pyramid. In fact, this is because of the important property of the defined total order over the indices of vertices where at the base level each non-surviving vertex only can be contracted into its right or down neighborhood vertex.

4. Fast Labeled Spanning Tree (FLST)

A CC in a binary graph pyramid is represented by a single surviving vertex at the top level of the pyramid. This vertex is the root of the tree spanning its receptive field at the base level [21]. In [22] it was shown that the combination of two (or more) successive reductions in an equivalent weighting function allows to calculate any level of the pyramid directly from the base. Kropatsch in [21] introduced the Equivalent Contraction Kernels (ECK) in the irregular graph pyramid and it was later used [23] in the minimum spanning tree (MST) segmentation.

In the binary pyramid, every spanning tree of a CC is the MST because the contrast (weight) of the intra-CC edges is zero. To drive the spanning tree of a CC, the previous common methods [14, 16, 11] need to select the CKs in n iterations where n is the height of the pyramid. In contrast, in the proposed method we only need two steps of selecting the CKs. Moreover, the SCK function is performed locally over each vertex. This means that the CKs are selected with parallel complexity of $O(1)$. Note that, it is assumed there are sufficient processing elements available in order to do the parallel computations.

4.1. Independent Edges

To contract the CKs in a parallel manner, finding a set of independent edges plays the key role. Dependency of the edges differs based on what processing is going to be performed between a set of edges. In [19] two edges not sharing an endpoint are considered as *independent* edges. Using this definition all the CKs at the first selection can be contracted with parallel complexity bounded as follows:

$$O(log_2(\delta(CK))) \leq CKs\ contraction \leq O(log_3(\delta(CK))) \tag{10}$$

To determine the parallel complexity of contracting the CKs at the second step of selection, the dependencies between darts [6] is considered. Since in this step, each edge of the CK is a bridge at the base level, hence, there is an inter-CCs edge with a σ-relation incident to this edge. Therefore, all the CKs at the second selections are independent of each other and they will be contracted in parallel complexity $O(1)$.

5. Applications

To highlight the usefulness of the proposed method, two main applications are presented. In both application the parallel complexity is $O(log(N))$ in a $N \times N$-size input binary image.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

5.1. Connected Component Labeling

Connected Component Labeling (CCL) is a fundamental task in analyzing binary images [24] where background and foreground are denoted by zero and one, respectively. A connected region is a group of pixels where all pairs of pixels are connected together. The role of the CCL is to assign a unique label to each CC. Common methods of CCL [24, 25, 26] are linear; i.e., they search the binary image row by row in the raster-scan fashion. In contrast, using a hierarchical structure, within the bottom-up construction each pixel reaches its single surviving pixel (super-pixel) at top of the pyramid in a logarithmic number of steps. At this top-level of the pyramid, each of the super-pixels receives its unique label Idx. Afterwards, through the top-down propagation the vertices of the lower levels inherit the labels from the higher levels until all the pixels at the receptive field (base level of the pyramid) received their labels.

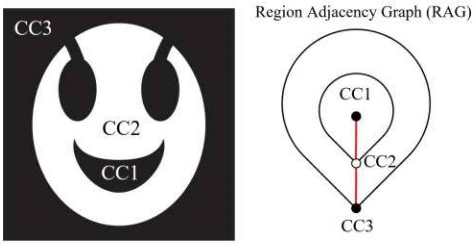


Figure 7: CCL by //ACC method.

The hierarchical method is called Parallel Pyramidal Connected Component (//ACC[2]) where the details can be found in [19]. The //ACC not only does the CCL task but also preserves the topological relations between the CCs. Fig. 7 shows how the //ACC encodes inclusion relationships between three CCs. Table 1 [19] shows the execution time of the //ACC method over three different categories of binary images; Random, MRI and Finger-print images. In addition, the execution time of the algorithm over different image-size is compared to the state-of-the-art methods; Spaghetti_RemSP, BBDT_RemSP, SAUF_UF, in [27]. The results in Fig. 8 encourage the //ACC method should be used in large images including more than one million pixels.

Table 1: Results for the different categories from (YACCLAB[28]).

Database Type	Random	MRI	Finger-print
size of Images	128×128	256×256	300×300
Num. of Images	89	1170	962
mean time (ms)	0.098	1.643	2.317
worst time (ms)	0.127	2.973	3.518

5.2. Distance Transform

The distance transform (DT) is another important fundamental operation that is applied to the binary image [1]. It is employed in a broad range of applications containing template matching [29, 30], image registration [31], map matching robot self-Localization [32], skeletonization [33], Line Detection in Manuscripts [34], Weather Analysis and Forecasting [35], etc. After applying the DT to a binary image, the result of the transform is a new gray-scale image whose foreground 1 pixels have intensities representing the minimum distance from the background 0 pixels.

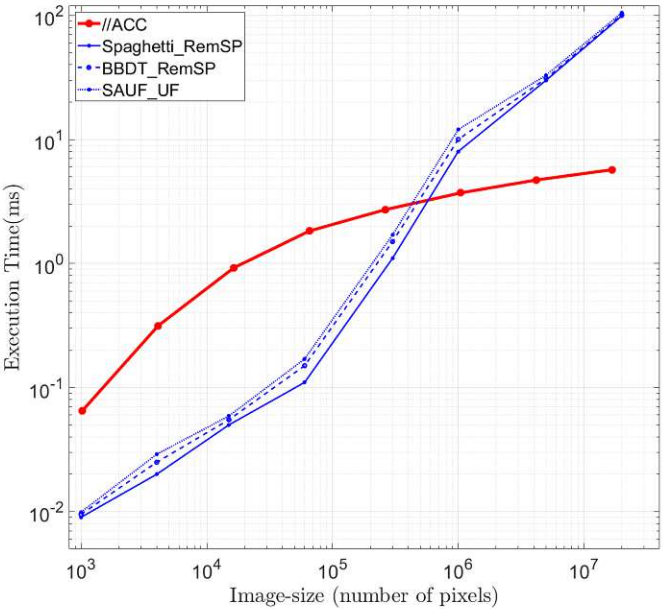


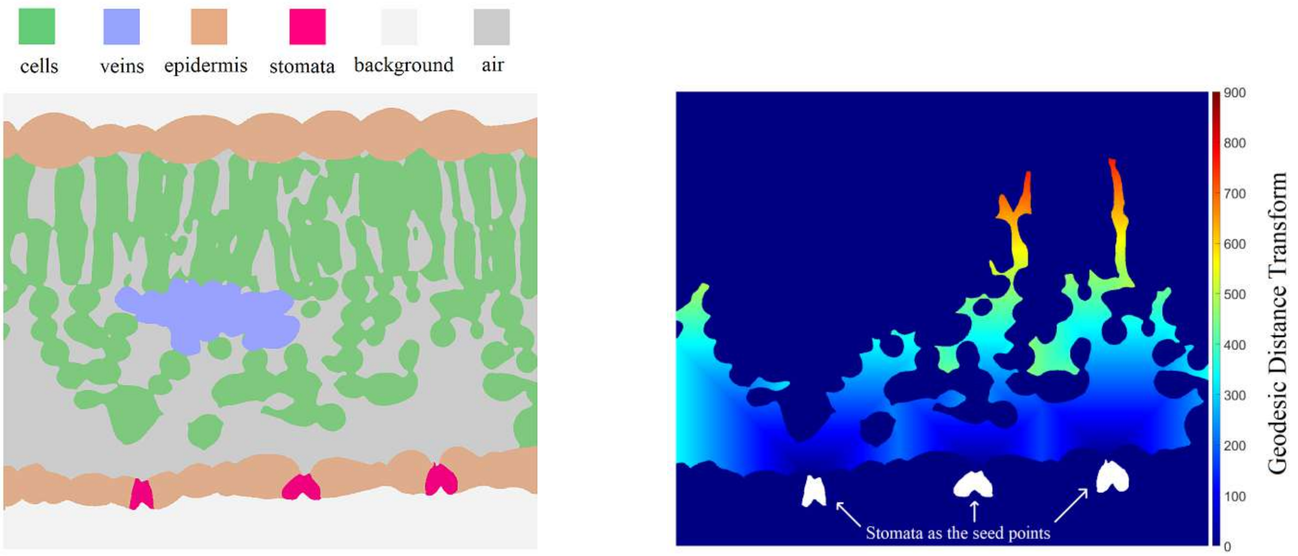
Figure 8: Illustration of the execution time (ms) over different image-sizes

In order to compute the DT, the common methods [1, 36], propagate the distances in linear sequential time. By contrast, using the hierarchical structure the distances can be propagated by a set of power-of-two numbers [20] where the parallel complexity is reduced into the logarithmic-time. The computation of DT with lower complexity makes the pyramid as a useful tool in analysing large binary images. In particular, currently we are working on the *Water's gateway to heaven* project[3] dealing with high-resolution X-ray micro-tomography (μCT) and fluorescence microscopy. The size of the images is more than 2000 in each of 3 dimensions where we use the saddle points of the DT to separate cells, which are visually difficult to be separated.

In the mentioned project above the input image is a labeled 2D cross slice of a leaf scan where it has six different labels illustrating different regions inside the leaf (Fig. 9a). The task of stomata is to control the amount of CO_2 that is entering the leaf. In order to do the photosynthesis, the CO_2 propagates through the airspace inside the leaf to reach the cells where it combines with water and sunlight. To model the procedure of the gas exchange in the leaf [37], we compute the geodesic distance transform (GDT) from the stomata through the airspace (Fig. 9b) to find out how long it takes to reach the necessary CO_2 concentration [20]. The use of pyramids would enormously speed up the computations of the DT in the large images of the project.

---

[2] It is pronounced pac where the // and A stand for parallel and pyramidal.
[3] https://waters-gateway.boku.ac.at/



(a) The labeled cross slice of a leaf

(b) Computing the Geodesic Distance Transform (GDT)

Figure 9: Computing the Geodesic Distance Transform in a multi-labeled image [20].

6. Conclusion

The paper presents a fast parallel method to select the equivalent contraction kernels in the irregular pyramid of a binary input image. It was shown that the first step of selecting the contraction kernels (CKs) at the base level is done with parallel complexity $O(1)$. These CKs are contracted with parallel $O(log(\delta))$ complexity where the δ is the diameter of the maximum connected component (CC) in the neighborhood graph of the image. By detecting the redundant edges (RE) the selection of CKs is performed in one parallel step. By defining the independent set of edges, we proved that all the selected CKs at the second step of selection are contracted in parallel complexity $O(1)$. The Fast labeled spanning tree (FLST) of the CCs is produced with parallel complexity $O(log(\delta))$. Using the total order there is no random processing in construction of the pyramid and the resulting FLST is unique.

In addition, it was shown by employing the proposed FLST, that the fundamental operations in analyzing the binary image can be performed in lower parallel complexity. In particular, two main operations, connected component labeling (CCL) and distance transform (DT), were presented in detail. Finally, we presented how the proposed method can be useful in processing of the large images in practical real applications. For future works we plan to compute 3D distance transform in order to study the diffusion in the air space within a leaf.

Conflict of Interest The authors declare no conflict of interest.

Acknowledgment We acknowledge the Paul Scherrer Institut, Villigen, Switzerland for provision of beamtime at the TOMCAT beamline of the Swiss Light Source and would like to thank Dr. Goran Lovric for assistance. This work was supported by the Vienna Science and Technology Fund (WWTF), project LS19-013, and by the Austrian Science Fund (FWF), projects M2245 and P30275.

References

[1] A. Rosenfeld, J. L. Pfaltz, "Sequential operations in digital picture processing", *Association for Computing Machinery*, vol. 13, no. 4, p. 471–494, 1966.

[2] P. Meer, "Stochastic image pyramids", *Computer Vision, Graphics, and Image Processing*, vol. 45, no. 3, pp. 269–294, 1989.

[3] Z. Pizlo, *Problem Solving, Cognitive Mechanisms and Formal Models*, Cambridge University Press, 2022.

[4] M. C. Potter, B. Wyble, C. E. Hagmann, E. S. McCourt, "Detecting meaning in rsvp at 13 ms per picture", *Attention, Perception, & Psychophysics*, vol. 76, no. 2, pp. 270–279, 2014, doi: 10.3758/s13414-013-0605-z.

[5] J. A. Feldman, D. H. Ballard, "Connectionist models and their properties", *Cognitive science*, vol. 6, no. 3, pp. 205–254, 1982.

[6] M. Banaeyan, W. G. Kropatsch, "Pyramidal connected component labeling by irregular graph pyramid", "2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA)", pp. 1–5, 2021, doi:10.1109/IPRIA53572.2021.9483533.

[7] J.-M. Jolion, A. Rosenfeld, *A pyramid framework for early vision: multiresolutional computer vision*, vol. 251, Springer Science & Business Media, 2012.

[8] W. G. Kropatsch, "Building irregular pyramids by dual graph contraction", *IEE-Proc. Vision, Image and Signal Processing*, vol. Vol. 142, no. No. 6, pp. pp. 366–374, 1995, doi:10.1049/ip-vis:19952115.

[9] W. G. Kropatsch, H. Macho, "Finding the structure of connected components using dual irregular pyramids", "Cinquième Colloque DGCI", pp. 147–158, LLAIC1, Université d'Auvergne, ISBN 2-87663-040-0, 1995.

[10] Y. Haxhimusa, W. G. Kropatsch, "Segmentation graph hierarchies", "Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshops on SSPR 2004 and SPR 2004", vol. LNCS 3138, pp. 343–351, Springer, Berlin Heidelberg, New York, 2004.

[11] M. Cerman, I. Janusch, R. Gonzalez-Diaz, W. G. Kropatsch, "Topology-based image segmentation using LBP pyramids", *Machine Vision and Applications*, pp. 1–14, 2016, doi:10.1007/s00138-016-0795-1.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek
WIEN Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

[12] R. Klette, *Concise computer vision*, vol. 233, Springer, 2014.

[13] R. Trudeau, *Introduction to Graph Theory*, Dover Books on Mathematics, Dover Pub., 1993.

[14] L. Brun, W. G. Kropatsch, "Hierarchical graph encodings", O. Lézoray, L. Grady, eds., "Image Processing and Analysis with Graphs: Theory and Practice", pp. 305–349, CRC Press, 2012.

[15] L. Brun, W. Kropatsch, "Introduction to combinatorial pyramids", "Digital and Image Geometry", pp. 108–128, Springer, 2001.

[16] F. Torres, W. G. Kropatsch, "Canonical encoding of the combinatorial pyramid", "Proceedings of the 19th Computer Vision Winter Workshop", pp. 118–125, 2014.

[17] M. Banaeyan, D. Batavia, W. G. Kropatsch, "Removing redundancies in binary images", "International Conference on Intelligent Systems and Patterns Recognition (ISPR), Hammamet, Tunisia, March 24-25, 2022", pp. 221–233, Springer, 2022, doi:10.1007/978-3-031-08277-1_19.

[18] B. A. Davey, H. A. Priestley, *Introduction to lattices and order*, Cambridge university press, 2002.

[19] M. Banaeyan, W. G. Kropatsch, "Parallel $O(log(n))$ computation of the adjacency of connected components", "International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI), Paris, France, June 1-3, 2022", pp. 102–113, Springer, 2022, doi:10.1007/978-3-031-09282-4_9.

[20] M. Banaeyan, C. Carratù, W. G. Kropatsch, J. Hladůvka, "Fast distance transforms in graphs and in gmaps", "IAPR Joint International Workshops on Statistical Techniques in Pattern Recognition (SPR 2022) and Structural and Syntactic Pattern Recognition (SSPR 2022), Montreal, Canada, August 26-27, 2022", p. in print, 2022.

[21] W. G. Kropatsch, "Equivalent contraction kernels to build dual irregular pyramids", *Advances in Computer Science*, vol. Advances in Computer Vision, pp. pp. 99–107, 1997.

[22] P. J. Burt, E. H. Adelson, "The Laplacian pyramid as a compact image code", "Readings in computer vision", pp. 671–679, Elsevier, 1987.

[23] Y. Haxhimusa, A. Ion, W. G. Kropatsch, "Evaluating hierarchical graph-based segmentation", "18th International Conference on Pattern Recognition", vol. II, pp. 195–198, IEEE Comp.Soc., 2006, doi:10.1109/ICPR.2006.511.

[24] L. He, X. Ren, Q. Gao, X. Zhao, B. Yao, Y. Chao, "The connected-component labeling problem: A review of state-of-the-art algorithms", *Pattern Recognition*, vol. 70, pp. 25–43, 2017, doi:10.1016/j.patcog.2017.04.018.

[25] L. He, Y. Chao, K. Suzuki, "Two efficient label-equivalence-based connected-component labeling algorithms for 3D binary images", *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2122–2134, 2011, doi:10.1109/TIP.2011.2114352.

[26] U. H. Hernandez-Belmonte, V. Ayala-Ramirez, R. E. Sanchez-Yanez, "A comparative review of two-pass connected component labeling algorithms", "Mexican International Conference on Artificial Intelligence", pp. 452–462, Springer, 2011, doi:10.1007/978-3-642-25330-0_40.

[27] F. Bolelli, S. Allegretti, L. Baraldi, C. Grana, "Spaghetti labeling: Directed acyclic graphs for block-based connected components labeling", *IEEE Transactions on Image Processing*, vol. 29, pp. 1999–2012, 2020, doi:10.1109/TIP.2019.2946979.

[28] C. Grana, F. Bolelli, L. Baraldi, R. Vezzani, "YACCLAB - Yet Another Connected Components Labeling Benchmark", "2016 23rd International Conference on Pattern Recognition (ICPR)", pp. 3109–3114, Springer, 2016, doi:10.1109/ICPR.2016.7900112.

[29] S. Prakash, U. Jayaraman, P. Gupta, "Ear localization from side face images using distance transform and template matching", "2008 First Workshops on Image Processing Theory, Tools and Applications", pp. 1–8, IEEE, 2008, doi:10.1109/IPTA.2008.4743786.

[30] J. Lindblad, N. Sladoje, "Linear time distances between fuzzy sets with applications to pattern matching and classification", *IEEE Transactions on Image Processing*, vol. 23, no. 1, pp. 126–136, 2014, doi:10.1109/TIP.2013.2286904.

[31] B. Hill, R. A. Baldock, "Constrained distance transforms for spatial atlas registration", *BMC bioinformatics*, vol. 16, no. 1, pp. 1–10, 2015, doi:10.1186/s12859-015-0504-5.

[32] H. Sobreira, C. M. Costa, I. Sousa, L. Rocha, J. Lima, P. Farias, P. Costa, A. P. Moreira, "Map-matching algorithms for robot self-localization: a comparison between perfect match, iterative closest point and normal distributions transform", *Journal of Intelligent & Robotic Systems*, vol. 93, no. 3, pp. 533–546, 2019, doi:10.1007/s10846-017-0765-5.

[33] C. Niblack, P. B. Gibbons, D. W. Capson, "Generating skeletons and centerlines from the distance transform", *CVGIP: Graphical Models and Image Processing*, vol. 54, no. 5, pp. 420–437, 1992.

[34] M. Kassis, J. El-Sana, "Learning free line detection in manuscripts using distance transform graph", "2019 International Conference on Document Analysis and Recognition (ICDAR)", pp. 222–227, 2019, doi:10.1109/ICDAR.2019.00044.

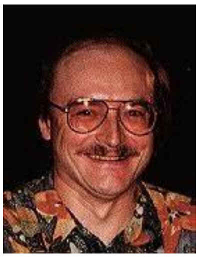
[35] D. Brunet, D. Sills, "A generalized distance transform: Theory and applications to weather analysis and forecasting", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 3, pp. 1752–1764, 2017, doi:10.1109/TGRS.2016.2632042.

[36] R. Fabbri, L. D. F. Costa, J. C. Torelli, O. M. Bruno, "2D euclidean distance transform algorithms: A comparative survey", *ACM Computing Surveys (CSUR)*, vol. 40, no. 1, pp. 1–44, 2008, doi:10.1145/1322432.1322434.

[37] M. Momayyezi, A. Borsuk, C. Brodersen, M. Gilbert, G. Théroux-Rancourt, D. Kluepfel, A. McElrone, "Desiccation of the leaf mesophyll and its implications for CO_2 diffusion and light processing", *Plant, Cell & Environment*, vol. 45, no. 5, pp. 1362 – 1381, 2022, doi:10.1111/pce.14287.

Copyright: This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see `https://creativecommons.org/licenses/by/4.0/`

MAJID BANAEYAN received his bachelor's degree in Computer Engineering from Isfahan University of Technology (IUT) in 2007. He has done his master's degree in Electrical Communication Engineering from Malek-Ashtar University of thechnology (MUT) in 2010. Currently he is doing his PhD in Informatics at Vienna University of Technology (TUWIEN). He is a member of Patter Recognition and Image Processing (PRIP) group under the supervision of Prof. Walter G. Kropatsch. He is working on hierarchical structure in pattern recognition, Irregular Graph Pyramid, Combinatorial and Generalized map.



WALTER G. KROPATSCH. From 1990-2021 he was full professor at TU Wien. He received his diploma degree in Technical Mathematics from the Technical University in Graz. He then moved to Grenoble, France to get the Maître d'Informatique from the University of Grenoble. His PhD in 1982 was on the Registration of Satellite Images with Maps. In 1984 he was invited by Prof. Azriel Rosenfeld to spend a year at the Center for Automation Research of the University of Maryland.

The creation of the first group in Austria dealing with pattern recognition and image processing in 1990 was jointly coordinated with the Austrian Association of Pattern Recognition (AAPR), that he initiated and led from 1984 until 1995. Under his leadership the AAPR became a member of the International Association of Pattern Recognition (IAPR) in which he held several leading positions, 2004-2006 he was its president. In 1996 he organized the main conference of the IAPR, the International Conference of Pattern Recognition in Wien, Austria. His scientific research focuses on pyramidal representations of images since his collaboration with Azriel Rosenfeld in 1984/85. The current graph-based pyramids follow similar concepts with the advantage that graphs are much more flexible data structures than the regular grids as currently used as architectures in deep learning. In his more than 400 scientific contributions many other concepts and applications have been addressed. He is currently senior editor of the journal of Electronic Imaging, and associate editor the journal of the Visual Computer and of several special issues in Pattern Recognition and Pattern Recognition Letter.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Paper E
Distance Transform in Parallel Logarithmic Complexity

Reprinted from [BK23a] with kind permission from Springer Nature.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Distance Transform in Parallel Logarithmic Complexity

Majid Banaeyan[a] and Walter G. Kropatsch[b]
Pattern Recognition and Image Processing Group, TU Wien, Vienna, Austria

Keywords: Distance Transform, Logarithmic Complexity, Irregular Pyramids, Parallel Processing.

Abstract: Nowadays a huge amount of digital data are generated every moment in a broad spectrum of application domains such as biomedical imaging, document processing, geosciences, remote sensing, video surveillance, etc. Processing such big data requires an efficient data structure, encouraging the algorithms with lower complexity and parallel operations. In this paper, first, a new method for computing the distance transform (DT) as the fundamental operation in binary images is presented. The method computes the DT with the parallel logarithmic complexity $O(log(n))$ where n is the maximum diameter of the largest foreground region in the 2D binary image. Second, we define the DT in the combinatorial map (CM) structure. In the CM, by replacing each edge with two darts a smoother DT with the double resolution is derived. Moreover, we compute n different distances for the nD-map. Both methods use the hierarchical irregular pyramid structure and have the advantage of preserving topological information between regions. The operations of the proposed algorithms are totally local and lead to parallel implementations. The GPU implementation of the algorithm has high performance while the bottleneck is the bandwidth of the memory or equivalently the number of available independent processing elements. Finally, the logarithmic complexity of the algorithm speeds up the execution and suits it, particularly for large images.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

1 INTRODUCTION AND MOTIVATION

The distance transform (DT) (Rosenfeld and Pfaltz, 1966) is a fundamental operation of many methods in pattern recognition and geometry. It is used in a wide range of applications such as skeletonization (Niblack et al., 1992), map matching robot self-Localization (Sobreira et al., 2019), image registration (Hill and Baldock, 2015), template matching (Prakash et al., 2008; Lindblad and Sladoje, 2014), Line Detection in Manuscripts (Kassis and El-Sana, 2019), Weather Analysis and Forecasting (Brunet and Sills, 2017), etc. The DT is applied to a binary image containing background and foreground regions. The result of the transform is a new gray-scale image whose foreground pixels have intensities representing the minimum distance from the background.

To compute the DT, the common algorithms (Rosenfeld and Pfaltz, 1966; Nilsson and Söderström, 2007; Fabbri et al., 2008) propagate the distances in linear-time $O(N)$, where N is the number of grid cells or pixels in a 2D binary image. In contrast, in this paper we propose a novel method that propagates the distances in an exponential way and computes the DT with $O(log(n))$ where n is the diameter of the largest connected component in the binary image. To this aim, we employ the hierarchical structure of the irregular graph pyramid (Kropatsch, 1995; Brun and Kropatsch, 2012; Banaeyan and Kropatsch, 2021). In addition, we define the DT over the combinatorial map (CM) that not only results in a finer resolution of the distance map but also provides different distances for different dimensions employing map-edit-distance (Combier et al., 2013) in analogy to the graph edit distance (Gao et al., 2010).

Currently we are working on the *Water's gateway to heaven* project[1] dealing with high-resolution X-ray micro-tomography (μCT) and fluorescence microscopy. The size of the images is more than 2000 in each dimension where we need the DT to separate cells, which are visually difficult to be separated. Therefore, fast computation of the DT with low complexity is required.

In this study, the proposed algorithm has logarithmic complexity and efficiently computes the *city block* (L_1 norm) distance metric. The next Section recalls the irregular graph pyramid and the combinatorial map. In Section 2, the new algorithm to com-

---

[a] https://orcid.org/0000-0001-8621-6424
[b] https://orcid.org/0000-0003-4915-4118

[1] https://waters-gateway.boku.ac.at/

Banaeyan, M. and Kropatsch, W.
Distance Transform in Parallel Logarithmic Complexity.
DOI: 10.5220/0011681500003411
In *Proceedings of the 12th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2023)*, pages 115-123
ISBN: 978-989-758-626-2; ISSN: 2184-4313
Copyright © 2023 by SCITEPRESS – Science and Technology Publications, Lda. Under CC license (CC BY-NC-ND 4.0)

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

pute the DT is presented. Section 3 introduces the DT over the combinatorial map and proposes a novel algorithm to compute the DT. Finally, Section 4 compares the execution of the proposed algorithm with the state-of-the-art.

1.1 Definitions

A digital image can be represented using a 4-adjacent neighborhood graph. Let $G = (V,E)$ be the neighborhood graph of image P where V corresponds to P and E relates neighboring pixels. Let the gray-value $g(v)$ of a vertex v be $g(p)$. The $contrast(e)$ is an attribute of an edge $e(u,v)$ where $u,v \in V$ and $contrast(e) = |g(u) - g(v)|$. In the binary images, the pixels (and corresponding vertices) have either of the two values 0 and 1. Similarly, the edge contrast has only two possible values 0 and 1.

In the neighborhood graph of the binary image, the edges with zero and one contrast are defined as **zero-edge**, e_0, and **one-edge**, e_1, respectively. Therefore, the edges of the graph are partitioned into $E = E_0 \cup E_1$ where $e_0 \in E_0$ and $e_1 \in E_1$.

1.1.1 Irregular Pyramids

Irregular pyramids (Kropatsch, 1995) are a stack of successively reduced smaller graphs where each graph is built from the graph below by selecting a specific subset of vertices and edges. Two basic operations are used to construct the pyramid: edge contraction and edge removal. In the edge contraction, an edge $e = (v,w)$ is contracted while its two endpoints, v and w, are identified and the edge is removed. The edges that were incident to the joined vertices will be incident to the resulting vertex after the operation. In edge removal, an edge is removed without changing the number of vertices or affecting the incidence relationships of other edges. In each level of the pyramid, the vertices and edges disappearing in the level above are called *non-surviving* and those appearing in the upper-level *surviving* ones.

Definition 1 (Contraction Kernel (CK))**.** *A CK is a tree consisting of a surviving vertex as its root and some non-surviving neighbors with the constraint that every non-survivor can be part of only one CK (Banaeyan and Kropatsch, 2022a).*

An edge of a CK is denoted by the directed edge and points towards the survivor.

1.1.2 Combinatorial Pyramids

A combinatorial pyramid is a hierarchy of successively reduced combinatorial maps (Brun and Kropatsch, 2003; Brun and Kropatsch, 2012). In the CM each edge is encoded by two half-edges where each half-edge is called a *dart*, $d \in \mathcal{D}$ where $\mathcal{D}$ is a finite set of darts. The CM encodes the edges around each vertex by using the α and the σ as an *involution* and a *permutation* on the set of $\mathcal{D}$, respectively. The σ encodes consecutive edges around the same vertex while turning counterclockwise. The clockwise orientation is denoted by σ^{-1}. The α provides a one-to-one mapping between consecutive darts forming the same edge such that $\alpha(\alpha(d)) = d$.

2 LOGARITHMIC DT USING THE IRREGULAR PYRAMID

In the linear algorithms (Nilsson and Söderström, 2007) the DT is propagated between one vertex (pixel) and its adjacent vertex (pixel) in each step of the propagation. Consider a 1D grid of N pixels aligning in a horizontal line. In order to propagate the DT from the most-left pixel to the most-right pixel, $N-1$ steps are needed. However, thanks to the hierarchical structure of the pyramid with logarithmic height, such propagation can be performed only in $log(N)$ steps as we will see in Section 2.3. In the pyramid, two vertices of a connected component that are not adjacent (and may be far from each other) at the base level, may become adjacent at the upper levels of the pyramid.

2.1 Initialization

To compute the DT, the first step is an initialization procedure where the endpoints of the E_1 receive $DT = 1$ and the remaining vertices receive the $DT = \infty$. Note that, the proposed algorithm computes the DT for both background and foreground regions simultaneously. This is the reason why in the initialization step we assign the $DT = 1$. The common algorithms for computation of the DT consider the background as a region with $DT = 0$. However, to convert the DT of the proposed algorithm to the common algorithms, it needs only to substitute the $DT = 0$ of the background pixels.

2.2 Selecting the CKs

Selecting the CKs is the main procedure in constructing the pyramid. To this aim, we use the proposed method in (Banaeyan et al., 2022). First, an index is assigned to each vertex. Using the total order set defined over the indices, each vertex has a unique integer index, Idx(.). Each non-surviving vertex selects one surviving vertex with maximum Idx of its neighborhood (Banaeyan et al., 2022):

$$v_s = \text{argmax}\{Idx(v_s)|\ v_s \in \mathcal{N}_0(v), |\mathcal{N}_0(v)| > 1\} \quad (1)$$

where

$$\mathcal{N}_0(v) = \{v\} \cup \{w \in V | e_0 = (v,w) \in E_0\} \quad (2)$$

The E_1 are the edges between two different CCs and the E_0 are the edges inside a CC. Since the E_1 have their $DT = 1$, we do not contract the E_1, and therefore, the CKs are selected only from the E_0. In addition, each edge $e_0 = (v,w)$ of the CKs has an orientation from v to w where the w has the largest index among the neighbors, $Idx(w) = \max\{Idx(u)|u \in \mathcal{N}_0(v)\}$.

2.3 Contracting the Selected CKs

In a CK, the adjacent edges are *dependent* and cannot be contracted at the same time. Two dependent edges by definition are adjacent edges sharing one endpoint. Those edges not sharing an endpoint are defined as *independent* edges.

Proposition 1. *A path of length N, can be contracted at maximum in* $[log_2(N)] + 1$ *steps.*

Proof. In the path of length N, every other edges have no endpoints in common and hence they are independent. As a result, such independent edges are contracted in one step. In the resulting induced graph, again, every other edges are independent and they can be contracted in one step. By iterating such procedure, the path of length N is contracted at maximum in $[log_2(N)] + 1$ steps. The number of required steps is equal to $log_2(N)$ when $N = 2^n$. ☐

Fig. 1 shows a 1D grid containing 16 vertices. The 1D grid is considered as the path with a length of 15 and it can be contracted in 4 steps. In each step the oriented edges are independent and they are contracted simultaneously. The priorities of the contractions are encoded by numbers 1 to 4 and the oriented edges are independent.

2.4 Logarithmic DT in 1D Grid

To compute the DT in 1D, the irregular pyramid is constructed in the bottom-up fashion. To this aim, the independent edges are identified based on the priorities of contractions. During the construction procedure, only the independent edges having two unknown DT at their endpoints are contracted. After the contractions, the vertices with known DT propagate their DT to their adjacent vertices at each level of the pyramid. Such propagation iterates until we reach to the top of the pyramid where there is no edge remaining for the contraction and all the vertices at this level have their own DT.

The next step is to traverse the irregular pyramid in the top-down procedure. In the top-down processing, each vertex inherits its DT to the same vertex at a level below. Afterward, the distances are propagated into their adjacent vertices. Such procedures iterate in each level until we reach to the base where all the vertices receive their own DT.
Note that the DT in each level of the irregular pyramid is propagated as follows:

$$D(v_i) = \min\{D(v_i), D(v_j) + |Idx(v_i) - Idx(v_j)| \\ |\ v_j \in \mathcal{N}(v_i)\} \quad (3)$$

Algorithm 1: Computing DT in a grid structure.

```
Input: Neighborhood Graph: $G = (V,E)$
Initialization: $DT = \infty, \forall v \in V$
    $DT = 1, \forall v \in E_1$
Propagating the distances to adjacent neighbors
Selecting the CKs (Bottom-up traversing)
While ($DT = \infty$ in the current level)
    Contracting the edges
    Propagating the distances to adjacent neighbors
end (Top of the Pyramid)
For ($j = L$ downto 1) (Top-down traversing)
    Imitate the DT from $L \rightarrow L-1$
    Propagating the distances to adjacent neighbors
end
```

Fig. 2 shows the computing of the DT in 1D grid by using the irregular pyramid. The Alg. 1 summarizes the steps of computing the DT in the grid structure by using the irregular pyramid.

2.5 Logarithmic DT in 2D Grid

Consider the binary image has M rows and N columns such that $(1,1)$ is the coordinate of the pixel $(p \in P)$ at the upper-left corner and (M,N) at the lower-right corner. The corresponding 4-adjacent neighborhood graph of the binary image has MN vertices. An index $Idx(.,.)$ of each vertex is defined (Banaeyan and Kropatsch, 2022b):

$$Idx : [1,M] \times [1,N] \mapsto [1, M \cdot N] \subset \mathbb{N} \quad (4)$$

$$Idx(r,c) = (c-1) \cdot M + r \quad (5)$$

where r and c are the row and column of the pixel, respectively. The Alg. 1 is used for computing the DT in the 2D grid as well. Here, The DT is propagated as

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

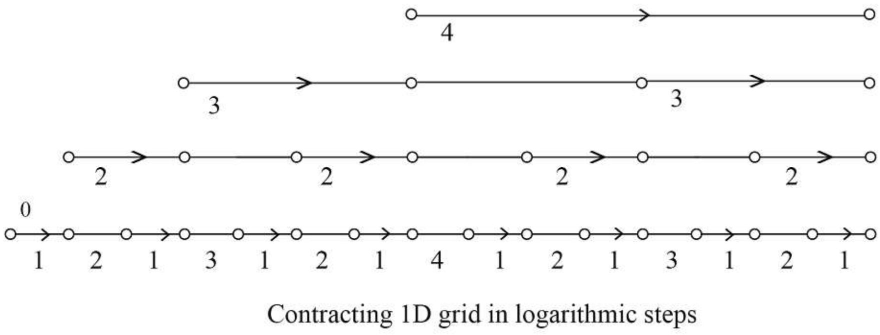
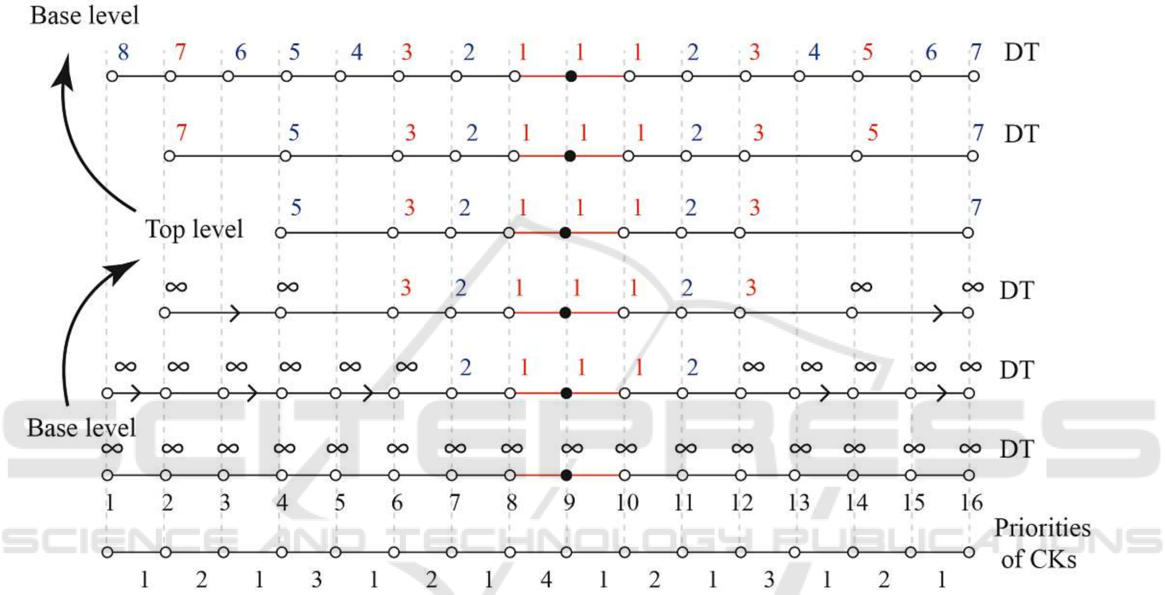


Figure 1: Edge contractions in the logarithmic way.



SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

Figure 2: Computing of the DT in 1D grid.

follows:

$$D(v_i) = \min\{D(v_i), D(v_j) + \begin{cases} 1 & if\ T = 1 \\ \frac{T}{M} & if\ T \neq 1 \end{cases} \} \tag{6}$$

where

$$v_j \in \mathcal{N}(v_i), \quad T = |Idx(v_i) - Idx(v_j)| \tag{7}$$

An example of computing the DT in a 2D binary image is shown in Fig. 3.

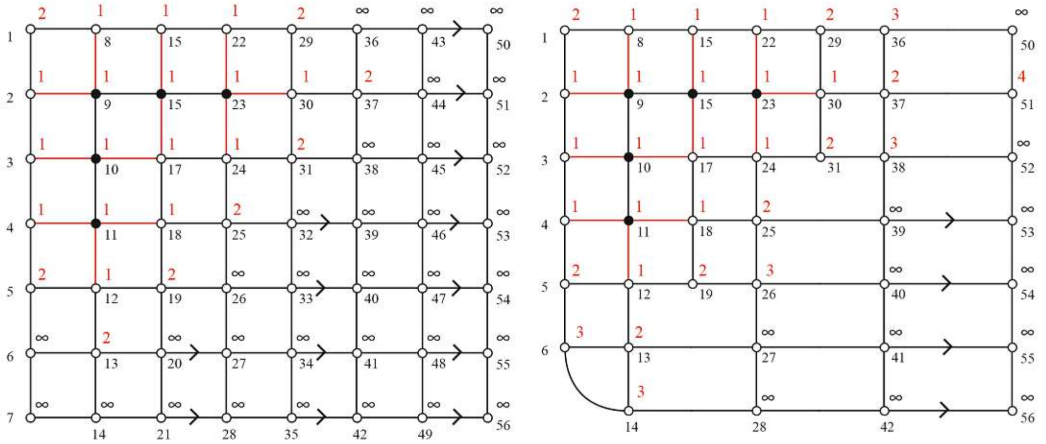
3 DEFINING THE DT IN A COMBINATORIAL MAP

The distance transform [1] computes for every pixel/voxel of an image/object how far it is from the closest obstacle, or boundary, or background. Different metrics can be used. In a topological data structure like a graph, a combinatorial map (Lienhardt, 1991), or a generalized map (Sansone et al., 2016) often the shortest path between the obstacle/boundary and a given point is used.

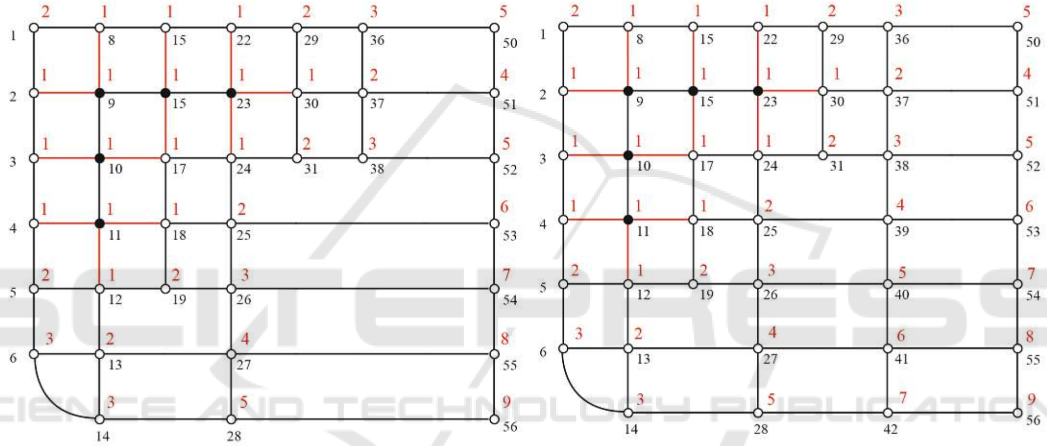
Let $(\mathcal{D}, \alpha, \sigma)$ denote a two-dimensional combinatorial map (2map). There are two versions of distance transform on a 2map. One considers the edges $\alpha^*(d)$ as a unit and counts the number of edges to follow as the distance. This corresponds to the distance in graphs. The alternative considers the darts $d \in \mathcal{D}$ as a unit and the following neighbors for propagating distances:

$$\Gamma_{2map}(d) = \{\alpha(d), \sigma(d), \sigma^{-1}(d)\} \tag{8}$$

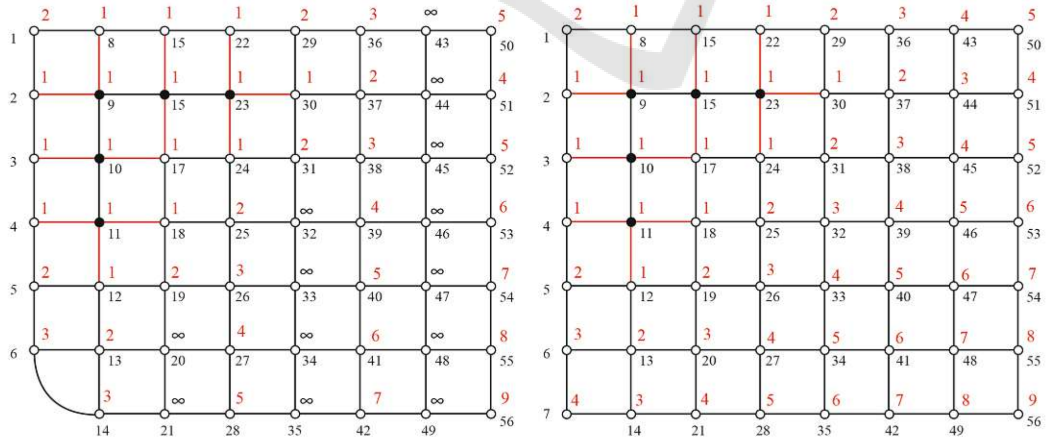
In the combinatorial map, each edge is replaced by two darts. Therefore, computing the DT for darts provides double resolution for the resulting distance map. Moreover, for every dimension $1, \ldots, n$ we receive one distance, the distance through the highest dimension n, and the (larger) geodesic distance along the bounding $i - cell$, $0 < i < n$. This characterizes more of a



(a) Initialization and Propagation of DT (b) Contractions and Propagation of DT



(c) Top of the pyramid (d) Copy and Propagation of DT



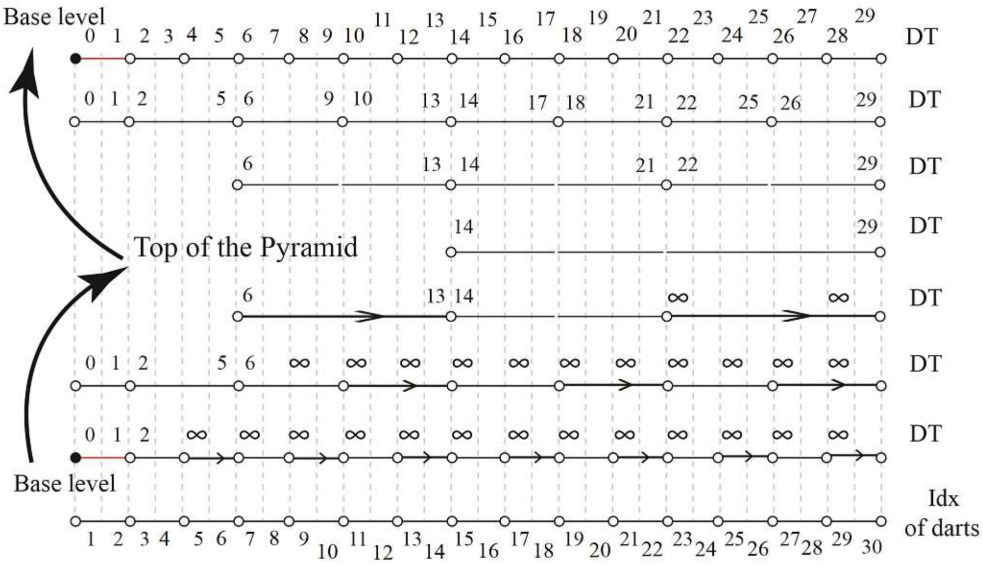
(e) Copy and Propagation of DT (f) Base of the pyramid

Figure 3: Computing of the DT in 2D grid.

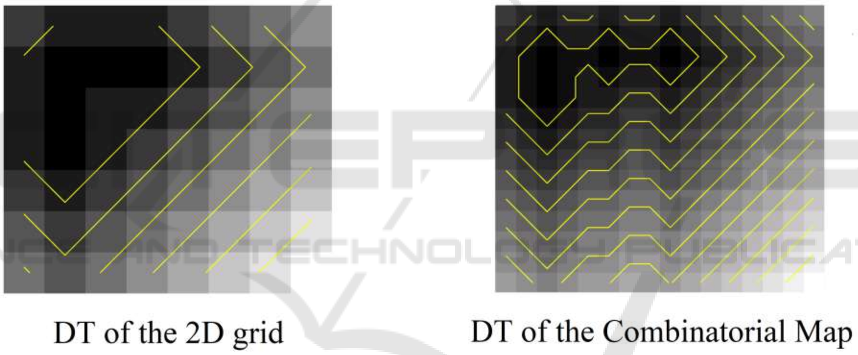
SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.



(a) Computing of the DT in 1D combinatorial map.



SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

(b) DT in the 2D grid and in the CM of Fig. 3.

Figure 4: Computing the DT in combinatorial map.

shape than just the highest dimension. In addition, it is a sort of map-edit-distance (Combier et al., 2013) in analogy to the graph edit distance (Gao et al., 2010).

3.1 Logarithmic DT in a 1D Combinatorial Map

To compute the DT in the CM, a similar algorithm to Alg. 1 can be used but with two modifications. First, the unique indices are defined for darts instead of vertices. Second, in each step, we propagate distances by α- and σ-propagation. The α-propagation of the DT is performed as follows:

$$D(d_i) = \min\{D(d_i), D(\alpha(d_i)) + |Idx(d_i) - Idx(\alpha(d_i))|\,\} \quad (9)$$

Note that, during the contraction of the $e = (d, \alpha(d))$, the $Idx(\sigma(d))$ of the contracted dart is updated after each contraction as follows:

$$Idx(\sigma(d)) = Idx(\alpha(d)) \quad (10)$$

The σ-propagation is performed as follows:

$$D(d_i) = \min\{D(d_i), D(\sigma(d_i)) + 1, D(\sigma^{-1}(d_i)) + 1\} \quad (11)$$

Fig. 4a shows an example of computing the DT in 1D CM. In constructing the pyramid in the bottom-up procedure, first, the α-propagation and then the σ-propagation are performed. In contrast, in the top-down procedure, they are performed the other way around. The steps of the algorithm are shown in Alg. 2 and Fig. 4b displays the finer resolution of the DT in comparison with the DT over 2D grid.

Table 1: Execution (*ms*) of proposed Logarithmic DT, MeijsterGPU and FastGPU.

Image-size	Mit. Log DT	MRI Log DT	Ran. Log DT	Ran. MeijsterGPU	Ran. FastGPU
256×256	0.0953	0.1209	0.0645	3.8524	1.7844
512×512	0.410	0.7310	0.3152	14.2005	4.2309
1024×1024	2.6308	5.1501	0.9364	25.8475	12.4668
2048×2048	4.1088	8.9506	1.8577	110.7817	44.9560

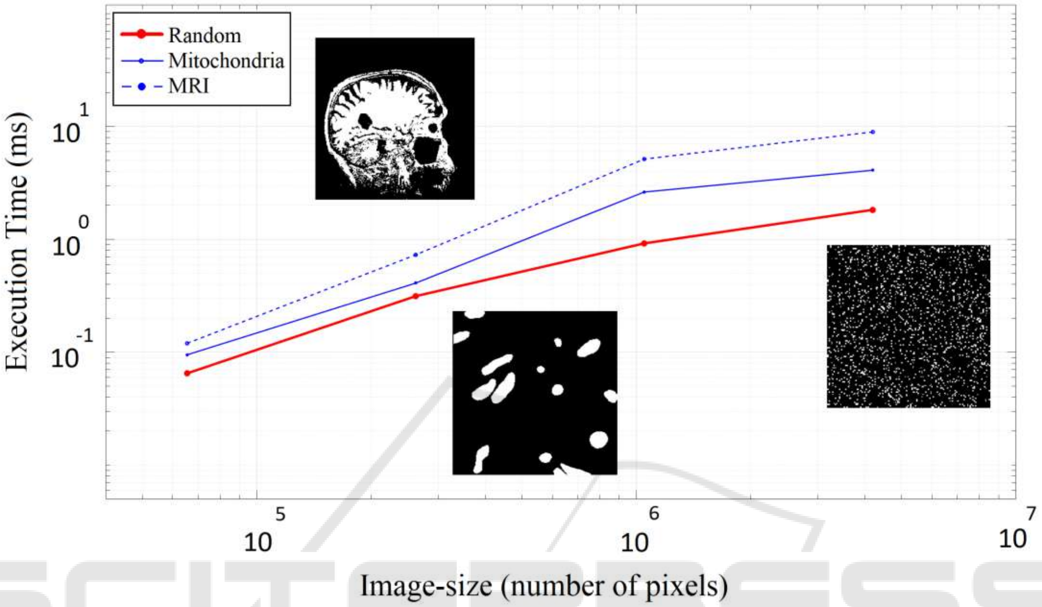


Figure 5: The proposed logarithmic DT over different images.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Algorithm 2: Computing DT in the 1D Combinatorial Map.

```
Input: $CM = (D, \alpha, \sigma)$
Initialization: $DT = \infty,\ \forall d \in D$
$DT(\sigma^*(d)) = 0,\ \forall d \in E_1$
$\alpha$- and $\sigma$-propagation to adjacent neighbors
Selecting the CKs (Bottom-up traversing)
While ($DT = \infty$ in the current level)
Contracting the edges with two unknown DT
$\alpha$- and $\sigma$-propagation to adjacent neighbors
end (Top of the Pyramid)
For ($j = L$ downto $1$) (Top-down traversing)
Imitate the DT from $L \rightarrow L-1$
$\sigma$- and $\alpha$-propagation to adjacent neighbors
end
```

4 COMPARISONS AND RESULTS

To highlight the advantages of the proposed logarithmic algorithm, we compare the execution times with two CUDA-based Implementations: MeijsterGPU and FastGPU in (de Assis Zampirolli and Filipe, 2017). Simulations use MATLAB software employing CPU with AMD Ryzen 7 2700X, 3.7GHz, and NVIDIA GeForce GTX 2080 TI that run over three different categories of images: Random, Mitochondria, and MRI. Table. 1 displays the outcome of the implementations. The first column shows the image size. The next three columns show the execution times (ms) of the proposed logarithmic DT (Log DT) in the three different classes of images. The last two columns show the execution time by the other two methods. Fig. 5 compares the results of the logarithmic algorithm between the different classes. Since the Random images contain smaller foreground objects than the other classes, they are executed faster. In Fig. 6 the logarithmic method is compared to MeijsterGPU and FastGPU methods. The logarithmic DT is not only significantly faster than the other ones but also has much higher performance dealing with larger images. Note that all operations and processes in the proposed algorithms are local and independent. Therefore, each available thread of the GPU in the shared memory is dedicated to each local process. The bottleneck of the algorithms is the capacity of the shared memory. Therefore, having sufficient independent processing elements the algorithms are fully parallel with logarithmic complexity.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

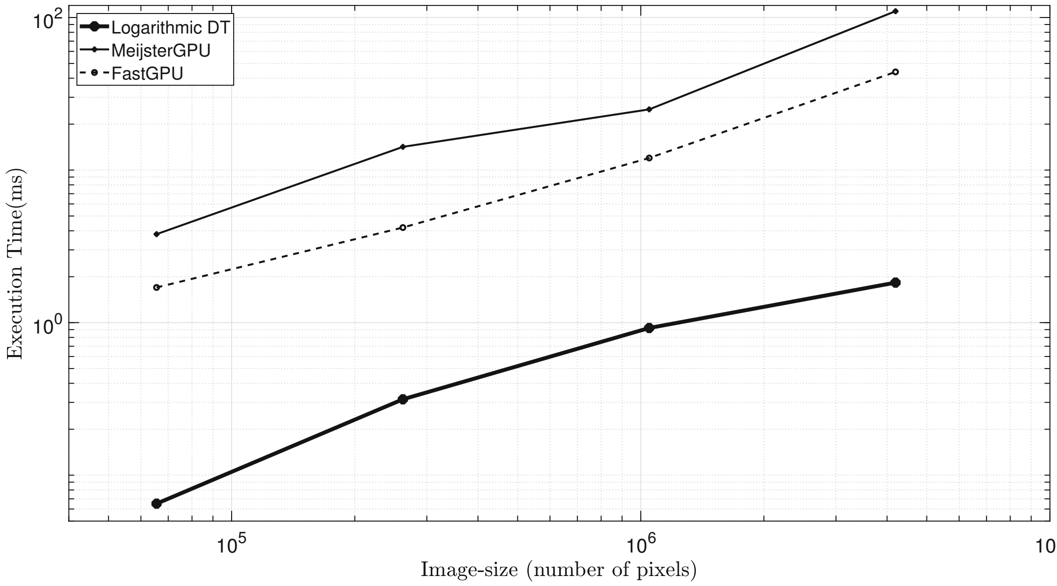


Figure 6: Comparison of the proposed algorithm with MeijsterGPU and FastGPU (de Assis Zampirolli and Filipe, 2017).

5 CONCLUSION

Distance transform (DT) computes how far inside a shape a point is located. In this paper, we study how the distances can be calculated in a discrete domain like a pixel grid and in combinatorial maps. Using the irregular pyramid we proposed a new algorithm that computes the DT in the logarithmic parallel complexity. Moreover, by defining the DT over the combinatorial maps, the smoother DT is calculated with double precision. Using the dart ordering, we proposed the logarithmic algorithm for computing the 1D combinatorial maps. However, the algorithm can be extended to higher n-dimensions. Finally, the practical results show that the algorithm with parallel logarithmic complexity notably decreases the execution time and makes it beneficial in particular for large images.

ACKNOWLEDGEMENTS

We acknowledge the Paul Scherrer Institut, Villigen, Switzerland for the provision of beamtime at the TOMCAT beamline of the Swiss Light Source and would like to thank Dr. Goran Lovric for his assistance. This work was supported by the Vienna Science and Technology Fund (WWTF), project LS19-013, and by the Austrian Science Fund (FWF), projects M2245 and P30275.

REFERENCES

Banaeyan, M., Batavia, D., and Kropatsch, W. G. (2022). Removing redundancies in binary images. In *International Conference on Intelligent Systems and Patterns Recognition (ISPR), Hammamet, Tunisia, March 24-25, 2022*, page 221–233. Springer.

Banaeyan, M. and Kropatsch, W. G. (2021). Pyramidal connected component labeling by irregular graph pyramid. In *5th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, pages 1–5.

Banaeyan, M. and Kropatsch, W. G. (2022a). Fast Labeled Spanning Tree in Binary Irregular Graph Pyramids. *Journal of Engineering Research and Sciences*, 1(10):69–78.

Banaeyan, M. and Kropatsch, W. G. (2022b). Parallel $O(log(n))$ computation of the adjacency of connected components. In *International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI), Paris, France, June 1-3, 2022*, page 102–113. Springer.

Brun, L. and Kropatsch, W. (2003). Combinatorial pyramids. In *Proceedings of International Conference on Image Processing*, volume 2, pages II–33. IEEE.

Brun, L. and Kropatsch, W. G. (2012). Hierarchical graph encodings. In Lézoray, O. and Grady, L., editors, *Image Processing and Analysis with Graphs: Theory and Practice*, pages 305–349. CRC Press.

Brunet, D. and Sills, D. (2017). A generalized distance transform: Theory and applications to weather analysis and forecasting. *IEEE Transactions on Geoscience and Remote Sensing*, 55(3):1752–1764.

Combier, C., Damiand, G., and Solnon, C. (2013). Map edit distance vs. graph edit distance for matching images. In *International Workshop on Graph-Based*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Representations in Pattern Recognition, pages 152–161. Springer.

de Assis Zampirolli, F. and Filipe, L. (2017). A fast cuda-based implementation for the euclidean distance transform. In *2017 International Conference on High Performance Computing Simulation (HPCS)*, pages 815–818.

Fabbri, R., Costa, L. D. F., Torelli, J. C., and Bruno, O. M. (2008). 2d euclidean distance transform algorithms: A comparative survey. *ACM Computing Surveys (CSUR)*, 40(1):1–44.

Gao, X., Xiao, B., Tao, D., and Li, X. (2010). A survey of graph edit distance. *Pattern Analysis and applications*, 13(1):113–129.

Hill, B. and Baldock, R. A. (2015). Constrained distance transforms for spatial atlas registration. *BMC bioinformatics*, 16(1):1–10.

Kassis, M. and El-Sana, J. (2019). Learning free line detection in manuscripts using distance transform graph. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 222–227.

Kropatsch, W. G. (1995). Building irregular pyramids by dual graph contraction. *IEE-Proc. Vision, Image and Signal Processing*, Vol. 142(No. 6):pp. 366–374.

Lienhardt, P. (1991). Topological models for boundary representation: a comparison with n-dimensional generalized maps. *Computer-aided design*, 23(1):59–82.

Lindblad, J. and Sladoje, N. (2014). Linear time distances between fuzzy sets with applications to pattern matching and classification. *IEEE Transactions on Image Processing*, 23(1):126–136.

Niblack, C., Gibbons, P. B., and Capson, D. W. (1992). Generating skeletons and centerlines from the distance transform. *CVGIP: Graphical Models and Image Processing*, 54(5):420–437.

Nilsson, O. and Söderström, A. (2007). Euclidian distance transform algorithms: A comparative study.

Prakash, S., Jayaraman, U., and Gupta, P. (2008). Ear localization from side face images using distance transform and template matching. In *2008 First Workshops on Image Processing Theory, Tools and Applications*, pages 1–8. IEEE.

Rosenfeld, A. and Pfaltz, J. L. (1966). Sequential operations in digital picture processing. *Association for Computing Machinery*, 13(4):471–494.

Sansone, C., Pucher, D., Artner, N. M., Kropatsch, W. G., Saggese, A., and Vento, M. (2016). Shape normalizing and tracking dancing worms. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 390–400.

Sobreira, H., Costa, C. M., Sousa, I., Rocha, L., Lima, J., Farias, P., Costa, P., and Moreira, A. P. (2019). Map-matching algorithms for robot self-localization: a comparison between perfect match, iterative closest point and normal distributions transform. *Journal of Intelligent & Robotic Systems*, 93(3):533–546.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

List of Figures

1.1 Growing Data volume and velocity [DK18]. 2
1.2 A small section of a 3D labeled image from the Water's Gateway to Heaven project [PVT$^+$22]. 3
1.3 Human problem-solving. Image is generated by AI, https://firefly.adobe.com 4
1.4 A segment of rods and cones in the human retina. Cones are represented by red dots, and Rods are depicted as blue dots. The green color illustrates bipolar cells that receive nerve signals from cones. Image courtesy of the National Eye Institute, National Institutes of Health (NEI/NIH). 5

2.1 Edge Operations [BK22b] 14
2.2 Two main operations in irregular graph pyramids. (a) Before applying an operation. (b), (c) after applying the operations [**?**]. (a) An edge e with its incident darts in the CM. (b) Removal operation, $G\backslash\{e\}$. (c) Contraction operation, $G/\{e\}$. 17

3.1 Generation of multiple edges and self-loops. 20
3.2 A non-invalid total order (left) fails to generate a spanning tree, while valid total orders (right) successfully yield spanning trees. 22
3.3 An example of a binary image showcasing a checkerboard pattern, where its corresponding neighborhood graph contains no redundant edges. 23
3.4 An example of the usual construction [BK00] of the irregular pyramid. . . 26
3.5 Construction of the irregular pyramid using the RtC method. The input graph at the base level is identical to Fig. 3.4. 27

4.1 Left: a set of independent edges and Right: a set of dependent darts (black color). Two red darts are independent. 31
4.2 Four distinct sets of independent edges. 31

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Appendix
Curriculum Vitae and Publications

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Majid Banaeyan Ph.D.
Favoritenstrasse 9 1040 Wien Austria

Email: majid@prip.tuwien.ac.at

Phone: +43 (1) 58801 - 18667

Education

• Ph.D. Institute of Visual Computing and Human-Centered Technische Universität Wien (TUWIEN), Vienna Austria, 2015 – 2024
 - Thesis: Redundant Edges and Parallel Algorithms in Irregular Graph Pyramids
 - Supervisor: Prof. Walter G. Kropatsch

• Visiting Research Scholar at Center of Machine Perception (CMP), Czech Technical University In Prague, Czech Republic, 2016 (June-September)
 - Research Title: 360º Multi-Camera Calibration
 - Supervisor: Prof. Vaclav Hlavac

• Master student Institute of Telecommunications, Technische Universität Wien (TUWIEN), Vienna Austria, 2014 – 2015

• M.Sc. Electrical Engineering Communication Systems, Malek-Ashtar University of Technology (MUT), Tehran Iran, 2007 – 2010
 - Thesis: Simulation and Comparison of Scene Matching Algorithms for Unmanned Airplane Vehicles in Spatial Domain
 - Supervisor: Prof. Hamid Soltanianzadeh

• B.Sc. Computer Engineering Hardware, Isfahan University of Technology (IUT), Isfahan Iran, 2003 – 2007
 - Thesis: Designing the dead-time for invertor’s input using FPGA
 - Supervisor: Assoc. Prof. Hamidreza Karshenas

Publications

- **M. Banaeyan,** W. G. Kropatsch, " Adapting a valid total vertex order to the geometry of a connected component", Pattern Recognition Letter, Volume ?, Issue ?, (submitted Jan 2024)
- **M. Banaeyan,** W. G. Kropatsch, "Distance Transform in Images and Connected Plane Graphs", Proceedings of International Conference on Pattern Recognition Applications and Methods (ICPRAM) – LNCS, volume 14547, PP. 19-32, Springer 2024.

- W. G. Kropatsch, **M. Banaeyan** "Controlling Topology Preserving Graph Pyramids" in "Emerging topics in Pattern Recognition and Artificial Intelligence", Pattern Recognition and Intelligent Systems, World Scientific **book,** World Scientific, 2024. (In print)
- **M. Banaeyan,** W. G. Kropatsch, "Reducing the Computational Complexity of the Eccentricity Transform of a Tree ", Proceedings of the 13th AIPR-TC15 International Workshop on Graph-Based Representation, Springer LNCS 2023, PP. 160-171.
- **M. Banaeyan,** W. G. Kropatsch, "Distance Transform in Parallel Logarithmic Complexity", Proceedings of the 12th International Conference on Pattern Recognition Applications and Methods - ICPRAM 2023, PP. 115-123, Lisbon, Portugal, SCITEPRESS, DOI: 10.5220/0011681500003411.
- **M. Banaeyan,** W. G. Kropatsch, "Fast Labeled Spanning Tree in Binary Irregular Graph Pyramids", In Hermann Bürstmayr, Andreas Gronauer, Andreas Holzinger, Peter M. Roth, and Karl Stampfer, editors, Proc. OAGM Workshop, Digitalization for Smart Farming and Forestry, pages 5–22. Technische Universität Graz, September 2022.
- **M. Banaeyan,** W. G. Kropatsch, " Redundant 1-cells in Multi-labeled 2-Gmap Irregular Pyramids", Journal of Engineering Research and Sciences, Volume 1, Issue 10, 69-78, 2022; DOI: 10.55708/js0110009.
- **M. Banaeyan,** C. Carratù, W. G. Kropatsch, and J. Hladůvka, "Fast Distance Transforms in Graphs and in Gmaps", IAPR Joint International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR), Montreal, Canada, PP. 193-202, August 26-27, Springer Nature, 2022.
- **M. Banaeyan,** W. G. Kropatsch, "Parallel O(log(n)) Computation of the Adjacency of Connected Components", 3rd International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI 2022), 1-3 June 2022, PP. 102-113, Paris-France, volume 13364. Springer LNCS, 2022.
- **M. Banaeyan,** D. Batavia, W. G. Kropatsch, "Removing Redundancies in Binary Images", 2nd International Conference on Intelligent Systems and Patterns Recognition (ISPR'2022), PP. 221-233, 24-26 March 2022, Hammamet-Tunisia, Springer LNCS, 2022.
- **M. Banaeyan,** W. G. Kropatsch, "Pyramidal Connected Component Labeling by Irregular Pyramid", 5th International Conference on Pattern Recognition and Image Analysis (IPRIA), PP. 1-5, 28-29 April 2021. IEEE doi: 10.1109/IPRIA53572.2021.9483533.
- **M.Banaeyan**, F. Moteabbed. "360º Calibration Tunnel for Around-view Monitoring System", 8th International Conference on New Solutions in Engineering, Information Science and Technology of the Century Ahead (ICIET), Dubai-UAE, 19th Apr 2021.
- **M.Banaeyan**, W. G. Kropatsch, R. Zamanshoar. "Pyramidal Connected Component Labeling of Image", 4th International Conference on Electrical, Electronic Engineering and Smart Grids, Tbilisi-Georgia, 17 December 2020.
- **M.Banaeyan.** "Model-Free Multi-Camera Calibration by Graph Pyramid using 360° Pattern", 4th International Conference on Electrical, Electronic Engineering and Smart Grids, Tbilisi-Georgia, 17 December 2020.
- **M. Banaeyan**, H. Huber, W. G. Kropatsch, and R. Barth. "A novel concept for smart camera image stitching." In Proceedings of the 21st Computer Vision Winter Workshop, Rimske Toplice, Slovenia, 3-5 Feb 2016.
- M. Kazemi, **M. Banaeyan** and A. Pourmohammad, "UAV Navigation Based on PIIFD/INS Method", International Journal of Computer Theory and Engineering, Vol. 4, No. 2, PP. 283-287, April 2012.
- Y.pourasad, H. Hassibi and **M. Banaeyan**, "Persian Characters Recognition Based on Spatial Matching," International Review on Computers and Software (I.RE.CO.S.), Vol. 6, No. 1, PP. 55-59, January 2011.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

- Y. pourasad, H. Hassibi and **M. Banaeyan**, "Farsi Font Recognition Based on Spatial Matching," 18th International Conference on Systems, Signals and Image Processing (IWSSIP-2011), Sarajevo, Bosnia and Herzegovina, June 16-18.
- **M. Banaeyan**, Neda B. Marvasti, and M. Kazemi, "Image Processing and Its Applications," Workshop of the 13th IEEE Electrical Engineering Student Conference (ISCEE 2010), Tarbiat Modares University, Tehran, Iran.
- **M. Banaeyan**, M. Kazemi and A. Pourmohammad, "Geometric Correction on IRS-P6 Images," In Proceedings of the 13th IEEE Electrical Engineering Student Conference (ISCEE 2010), Tarbiat Modares University, Tehran, Iran.
- **M. Banaeyan**, M. Kazemi and A. Pourmohammad, "Image Compression in Spatial Domain," proceeding of the 13th IEEE Electrical Engineering Student Conference (ISCEE 2010), Tarbiat Modares University, Tehran, Iran.
- **M. Banaeyan**, M. Kazemi and A.Pourmohammad, "Improving Image Compression in Spatial Domain Using Edge of Objects," In proceeding of the 13th IEEE Electrical Engineering Student Conference (ISCEE 2010), Tarbiat Modares University, Tehran, Iran.
- **M. Banaeyan** and Neda B. Marvasti, "Image ciphering using classic ciphering methods," In proceeding of the 11th IEEE Electrical Engineering Student Conference (ISCEE 2008), Zanjan University, Zanjan, Iran.

Honors and Awards

- Reviewer of American Journal of Artificial Intelligence, 2023-Present
- Program Committee: International Conference on Intelligent Systems & Pattern Recognition (ISPR), 2023
- Reviewer of IAPR International Conference on Discrete Geometry and Mathematical Morphology (DGMM), 2020
- Reviewer of International Conference on Intelligent Systems & Pattern Recognition (ISPR), 2020-Present
- The Best paper of 13th IEEE Electrical Engineering Student Conference (ISCEE 2010), Tarbiat Modares University Tehran Iran, 2010
- Reviewer Chairman of 11th, 12th, and 13th IEEE Electrical Engineering Student Conference (ISCEE), Image Processing Sessions, 2009-2011
- Supervisor of 62 Bachelor Theses, various Universities in Tehran Iran, 2010 - 2014

Work Experience

- Ph.D. Student & Project Assistant, Pattern Recognition and Image Processing Group (PRIP), TUWIEN, Vienna Austria, 2015 – 2024
  - Image and Graph Pyramids, Combinatorial Maps, n-Generalized Maps
  - 3D modeling and analyzing of high-resolution micro-CT images
  - Camera Calibration of 360° images in multi-camera systems

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

- University Lecturer, Shahr-e-Qods Campus and Abrar Institute of Higher Education, 2010 – 2014
  - Taught courses and managed projects in various subjects including Digital Image Processing, Microprocessors, Microprocessor Lab, Digital Communication systems, Signals and Systems, Digital Electronic, Linear Control Systems, C programming

- Electrical and Computer Engineer, Kara Pooyesh Tajhiz, Tehran Iran, 2010 – 2014
  - Developed image processing algorithms for Remote Sensing, Geometric and Radiometric corrections of satellite images, Visual navigation of UAVs, Keyword Spotting and OCR in document analysis

TU Bibliothek WIEN Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.