HAL
open science

Graphs and Binary Linear Programming for Natural Object Modeling in Computer Vision

Roaa Soloh

To cite this version:

Roaa Soloh. Graphs and Binary Linear Programming for Natural Object Modeling in Computer Vision. Optimization and Control [math.OC]. Normandie Université, 2022. English. NNT : 2022NORMLH12 . tel-03867801v2

HAL Id: tel-03867801

https://normandie-univ.hal.science/tel-03867801v2

Submitted on 16 Dec 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Normandie Université

THÈSE

Pour obtenir le diplôme de doctorat

Spécialité MATHEMATIQUES

Préparée au sein de l'Université Le Havre Normandie

En cotutelle internationale avec Université Libanaise , LIBAN

Graphs and Binary Linear Programming for Natural Object Modeling in Computer Vision

Présentée et soutenue par
ROAA SOLOH

Thèse soutenue le 22/09/2022 devant le jury composé de		
MME SAMIA BOUKIR	PROFESSEUR DES UNIVERSITES, INST POLYTECHNIQUE BORDEAUX	Rapporteur du jury
M. CYRIL FONLUPT	PROFESSEUR DES UNIVERSITES, ULCO - UNIVERSITE DU LITTORAL COTE D'OPALE	Rapporteur du jury
M. HASSAN ALABBOUD	ASSOCIATE PROFESSOR, UNIVERSITE LIBANAISE	Membre du jury Co-encadrant
M. ABDALLAH EL CHAKIK	ASSOCIATE PROFESSOR, BEIRUT ARAB UNIVERSITY	Membre du jury
MME NACIMA LABADIE	PROFESSEUR DES UNIVERSITES, UNIVERSITE TECHNOLOGIE TROYES (UTT)	Président du jury
M. ADNAN YASSINE	PROFESSEUR DES UNIVERSITES, Université Le Havre Normandie	Directeur de thèse
M. AHMAD SHAHIN	PROFESSEUR, UNIVERSITE LIBANAISE	Co-directeur de thèse

Thèse dirigée par ADNAN YASSINE (Laboratoire de Mathématiques Appliquées du Havre) et AHMAD SHAHIN (Laboratoire de Mathématiques Appliquées du Havre)

UNIVERSITÉ LE HAVRE NORMANDIE

MIIΣ

LMAH

THESE

Pour obtenir le grade de Docteur de Normandie Université
et le grade de Docteur de l'Université Libanaise

Spécialité : Mathématiques Appliquées, Informatique

École Doctorale Mathématiques, Information, Ingénierie des Systèmes, France
École Doctorale en Sciences et Technologie, Liban

Graphs and Binary Linear Programming for Natural Object Modeling in Computer Vision

Présentée et soutenue par
Roaa SOLOH

Thèse soutenue publiquement le 22 septembre 2022 devant le jury composé de		
Mme Samia BOUKIR	Professeur, Institut Polytechnique de Bordeaux (IPB)	Rapporteuse
M. Cyril FONLUPT	Professeur, Université du Littoral Côte d'Opale	Rapporteur
Mme Nacima LABADIE	Professeur, Université de Technologie de Troyes	Examinatrice
M. Abdallah EL CHAKIK	Maître de Conférences, Beirut Arab University (BAU) – Liban	Examinateur
M. Hassan ALABBOUD	Maître de Conférences – Université Libanaise – Tripoli – Liban	Co-Encadreur de thèse
M. Ahmad SHAHIN	Professeur – Université Libanaise – Tripoli – Liban	Directeur de Thèse
M. Adnan YASSINE	Professeur – Université Le Havre Normandie	Directeur de Thèse

Thèse dirigée par Adnan YASSINE, Laboratoire de Mathématiques Appliquées du Havre (LMAH), Université Le Havre Normandie – France et Ahmad SHAHIN, Laboratoire d'informatique et ses applications (LIA), Université Libanaise – Liban.

Etablissement **Ecole Doctorale** **Laboratoire**

بسم الله الرحمن الرحيم

وما توفيقي إلا بالله

لله تعالى، العظيم الجليـل الذي منَّ عليّ بإتمام هذا العمل
راجيةً منه سبحـانه أن يتقبّله خالصًا لوجهـه الكريم، وأن يكون جوابًا كافيًا لـ: "عن شبابي فيما أفنيته" ...

إلى حبيبي وقدوتي محمد ﷺ، سلكتُ طريقًـــا أبتغي فيه علمًـــا آملةً أن يسلك الله بي طريقًـــا إلى الجنة، مِصداقًـــا لحديثك الشريف...

إلى داعمي وسندي بعد الله، إلى أبي الغـــــالي،
حرصتَ على أن أستمر، علّمتني المجاهدة للوصول، أنرت عقلي بالعلم والعمل...

إلى مُلهمتي، إلى أمي الحبيبـــــة،
أنا صنيعتكِ، ومنكِ أستمد القدرة على المواجهة والصبـــــر...

إلى من غادرنا جسدًا، وبقيتْ وصيتهُ لي محفورةً في عقلي وفكري، ســـماحة المفتي خليـــــل الميس – رحمه الله -
كنتَ الفكر النيّر الذي يُضيء لي الطريق...

إلى أخي محمـــد، وزوجته ليـــــن
وجودكما كان فارقًا في استمراريتي وحبكمـــا أعانني في الطريـــق...

إلى أختي في الله نـــــداء،
كُنتِ مبعثَ قوةٍ لأكمل، دعمكِ بشتى الوسائـــل هوَّنَ عليَّ مشقة الطريـــق...

إلى كل من كان سندًا لي في هذه الرحـــلة، بالكلمة الطيبة والكثيـــر الكثيـــر من الدعـــــاء....

أرجو أن يكون هذا العمل المتـــواضع ذُخرًا لنا

رؤى

Acknowledgement

Finishing this thesis would not be achieved without the great blessings of **Allah**, the Most Gracious, and the Most Merciful. With the support, encouragement, and many prayers of many people Allah placed in my journey, I was able to have enough consistency and determination to work more and achieve better.

This thesis is the output of a journey, that started on 01/01/2019 and ended on 06/2022 as a collaboration between the Lebanese University - Lebanon and Le Havre University - France.

Foremost, I would like to express my deepest gratitude to Professor Adnan YASSINE, and Professor Ahmad SHAHIN for accepting me as their Ph.D. student, and assisting me in all the ways they can, to achieve this.

With his support, valuable comments, and the many positive vibes he gave me through the difficult times, I would like to say that words can not express my appreciation and gratitude for Doctor Hassan ALABBOUD.

I deeply thank the reporters and examiners of my thesis, Professor Samia BOUKIR, Professor Cyril FONLUPT, Professor Nacima LABADIE, and Doctor Abdallah EL CHAKIK.

Throughout this journey, I would like to appreciate all those who were great friends during my travels, Aicha, Hamdi, and Abdul-Aziz. Your support was important to me.

A special thanks to Professor Fawaz El-Omar, who played a big role in letting this journey happen.

My friends, those who proved that distance is not a matter of supporting, healing, and taking care mentally of me, Aya, Amina, Nadine, Hajar, and Sara. I owe you a big thanks for all that you did.

A great appreciation goes to both Doctor Mohamad Al-Abed, and Doctor Sanaa Kaddoura, for their continuous support and encouraging to do the best.

As a big supporter on this journey, I'm forever thankful to Nedaa that accompanied me through every step I did, encouraged me, and provided me with all the love and positive vibes.

The endless gratitude goes to my family, my beloved mother Joumana and father Khaled. Brother Mohamad, and sister-in-law Leen; their prayers, love, care, support,

patience, and understanding have lifted my spirits and given me the motivation to finish this thesis.

To all those who contributed to my thesis, providing an idea, a simple word to ease the difficult times, and they are many, thank you so much.

For me, pursuing this Ph.D. has been a difficult and life-changing journey. Without the help and advice I had received, I would not have been able to succeed. Thank you all so much for sticking by my side, believing in me, and working so hard to make my goal a reality.

Roaa

ABSTRACT

In the digital world, two-dimensional (2D) and three-dimensional (3D) shapes are important for representing real objects. Their applications span a wide range of fields, including medical, engineering, and security, etc... Considering the aspect that 2D and 3D models are widespread and because graphs are strong mathematical modeling tools used in a variety of computer science domains. We aim to represent our input as graphs to benefit from the highly meaningful representation. In this thesis, we conduct two parts.

The first part was related to the 3D models, where we addressed the problem of finding a superior one-to-one correspondence between the 3D models to obtain an optimal matching and retrieval. To do so, we detect feature points using the well-known 3D Harris detector, followed by proposing a combination of local shape descriptors to form a compact feature vector for the key points extracted that consist of Gaussian curvature, curvature index, and shape index. Then we model the matching problem as a combinatorial optimization problem solved using a brute-force approach and a Hungarian algorithm, comparing the efficiency between them. Our results were encouraging where despite the affine transformations between models, our descriptors were able to make efficient matching.

In the same framework working with these 3D models, we used Gaussian weight to represent our weighted graph and use the binary linear programming to segment our meshes into regions, where we tend to maximize the modularity between vertices, these regions are represented by a single point for each, this ends up for a graph matching problem between the models, treated as a combinatorial optimization problem. In this special work, we add a mean curvature as a descriptor in addition to the obtained descriptor, which leads to better results. To obtain the one-to-one correspondence we tend to minimize the cost function between the graphs. The significance of this work was to extract the descriptors for much fewer points than the 3D Harris detector, yet obtain well-matching results.

The idea of the second part came from the huge increase in the cancer disease diagnoses, in special cases the brain. And since early diagnosis will help to start treatment faster, in this part, we suggest making detection of the tumor generated automatically from Magnetic Resonance Images (MRIs) assisting the doctors. Using graph-based approach, our approach in this was to find an optimal way of preprocessing step to prepare the MRI which will be further represented as a weighted graph. By the use of an α-expansion move from the Boykov-Kolmogorov algorithm and a post-processing step to fully conduct the tumor.

By removing any artifacts from MRI and down-sampling it without affecting the resolution, an s/t graph cut was performed on the generated graph which represents the pixels as nodes and the difference of intensity as the weight of the edges. The cut obtained leads to an image segmentation, leading to some post-processing for

conduct the tumor only. We evaluate our framework on two datasets of nearly 400 2D MRIs, and the results show a high accuracy, specificity, and precision.

Keywords: Graphs, Optimization, Binary Linear Programming, Brain Tumor, Image Segmentation, Matching Problems, 3D objects, Descriptors, Classification & Retrieval.

Résumé

Dans le monde numérique, les formes bidimensionnelles (2D) et tridimensionnelles (3D) sont importantes pour représenter les objets réels. Leurs applications couvrent un large éventail de domaines, notamment la médecine, l'ingénierie, la sécurité, etc. Considérant l'aspect que les modèles 2D et 3D sont très répandus et parce que les graphes sont de puissants outils de modélisation mathématique utilisés dans une variété de domaines informatiques. Nous cherchons à représenter nos données d'entrée sous forme de graphes afin de bénéficier d'une représentation hautement significative. Dans cette thèse, nous menons deux parties.

La première partie était liée aux modèles 3D, où nous avons abordé le problème de la recherche d'une correspondance biunivoque supérieure entre les modèles 3D afin d'obtenir une correspondance et une récupération optimales. Pour ce faire, nous détectons les points caractéristiques à l'aide du célèbre détecteur 3D de Harris, puis nous proposons une combinaison de descripteurs de forme locaux pour former un vecteur de caractéristiques compact pour les points clés extraits, qui consiste en une courbure gaussienne, un indice de courbure et un indice de forme. Nous modélisons ensuite le problème de correspondance comme un problème d'optimisation combinatoire résolu à l'aide d'une approche de force brute et d'un algorithme hongrois, en comparant leur efficacité.

Nos résultats sont encourageants : malgré les transformations affines entre les modèles, nos descripteurs sont capables de réaliser une correspondance efficace. Dans le même cadre de travail avec ces modèles 3D, nous avons utilisé un poids gaussien pour représenter notre graphe pondéré et utiliser la programmation linéaire binaire pour segmenter nos mailles en régions, où nous tendons à maximiser la modularité entre les sommets, ces régions sont représentées par un seul point pour chacune, ce qui aboutit à un problème de correspondance de graphe entre les modèles, traité comme un problème d'optimisation combinatoire.

Dans ce travail spécial, nous ajoutons une courbure moyenne comme descripteur en plus du descripteur obtenu, ce qui conduit à de meilleurs résultats. Pour obtenir la correspondance biunivoque, nous tendons à minimiser la fonction de coût entre les graphes. L'intérêt de ce travail était d'extraire les descripteurs pour beaucoup moins de points que le détecteur 3D de Harris, tout en obtenant de bons résultats de correspondance.

L'idée de la deuxième partie est venue de l'augmentation considérable du nombre de diagnostics de maladies cancéreuses, en particulier dans le cerveau. Et comme un diagnostic précoce permet de commencer le traitement plus rapidement, nous proposons dans cette partie de faire de la détection de la tumeur générée automatiquement à partir d'images à résonance magnétique (IRM) une aide pour les médecins. En utilisant l'approche basée sur les graphes, notre approche a été de trouver une manière optimale d'étape de prétraitement pour préparer l'IRM qui sera ensuite représenté

comme un graphe pondéré.

En utilisant un mouvement d'expansion de l'algorithme de Boykov-Kolmogorov et une étape de post-traitement pour conduire complètement la tumeur. En supprimant tous les artefacts de l'IRM et en la sous-échantillonnant sans affecter la résolution, une coupe de graphe s/t a été effectuée sur le graphe généré qui représente les pixels comme des nœuds et la différence d'intensité comme le poids des bords. La coupe obtenue conduit à une segmentation de l'image, conduisant à un post-traitement pour conduire la tumeur uniquement. Nous évaluons notre cadre sur deux jeux de données de près de 400 IRM 2D, et les résultats montrent une exactitude, une spécificité et une précision élevées.

Mots-clés : Graphes, optimisation, programmation linéaire binaire, tumeur cérébrale, segmentation d'image, problèmes de correspondance, objets 3D, descripteurs, classification et récupération.

Author's Publications

Published Journal

[SCA+22] Soloh, R., El Chakik, A., Alabboud, H., Shahin, A. and Yassine, A. (2022) 'New descriptors' combination for 3D mesh correspondence and retrieval', Int. J. Computational Vision and Robotics, Vol. 12, No. 4, pp.377–396.

Published International Conference

[SEA+] Soloh, R., El Chakik, A., Alabboud, H., Shahin, A. and Yassine, A.(2021) '3d Mesh Matching Using Surface Descriptor and Integer Linear Programming', EJONS 12th INTERNATIONAL CONGRESS ON MATHEMATICS, ENGINEERING AND NATURAL SCIENCES, pp.338-347.

Submitted Journal Paper

Soloh, R., Alabboud, H., El Chakik, A., Shahin, A. and Yassine, A.(2022) "Brain Tumor Segmentation based on α-expansion graph cut". International Journal of Imaging Systems and Technology.

Contents

Acknowledgement i

Abstract iii

Résumé v

Author's Publications vii

Table of Contents viii

List of Figures ix

List of Tables xii

1 Introduction 1
- 1.1 Motivation . . . 2
  - 1.1.1 3D shape matching and retrieval . . . 4
  - 1.1.2 Medical Imaging . . . 4
- 1.2 Conducted Research Questions . . . 4
- 1.3 Organization of the thesis . . . 5
- 1.4 French Introduction . . . 6
  - 1.4.1 Motivation . . . 6
  - 1.4.2 Correspondance et récupération de formes 3D . . . 7
  - 1.4.3 Imagerie médicale . . . 7
  - 1.4.4 Questions de recherche traitées . . . 8

2 Preliminary Notions 9
- 2.1 Graphs . . . 10
  - 2.1.1 Basic Definitions and Notations . . . 10
  - 2.1.2 Graph matching . . . 13
- 2.2 Optimization . . . 14
  - 2.2.1 Continuous optimization . . . 15
  - 2.2.2 Discrete optimization . . . 15
  - 2.2.3 Detailed Discrete optimization . . . 15
- 2.3 Two and Three-Dimensional Models . . . 17

3 State of the art 20
- 3.1 3D objects matching and classification . . . 21
  - 3.1.1 3D Objects Detectors and Descriptors . . . 21
  - 3.1.2 3D Objects Matching, and Classification . . . 26
- 3.2 Brain Tumor Segmentation . . . 28
  - 3.2.1 Brain Imagining Modality . . . 29

3.2.2 Approaches for Brain Tumor Segmentation 32

4 New Descriptors' Combination for 3D Mesh Correspondence and Retrieval 37
4.1 Introduction 38
4.2 Proposed Approach 39
4.2.1 Interest Point Detection 39
4.2.2 Feature Descriptor Formulation 40
4.2.3 Feature Matching 42
4.3 Experimental Results 44
4.3.1 Interest Point Detection Experiments 44
4.3.2 Matching Experiments 46
4.4 Conclusion 49

5 3D mesh matching using surface descriptor and integer linear programming 51
5.1 Introduction 52
5.2 Proposed Framework 52
5.2.1 Descriptor map estimation 55
5.2.2 Feature Regions Detection 57
5.2.3 3D Surface Matching 59
5.3 Experimental data analysis and results 60
5.4 Conclusion 65

6 Brain Tumor Segmentation Based on α - Expansion Graph Cut 66
6.1 Introduction 67
6.2 Related Work 69
6.3 Methodology 72
6.3.1 Materials 72
6.3.2 Methods 73
6.4 Implementation 74
6.4.1 Configuration and Preprocessing Step 75
6.4.2 Alpha-expansion graph cut 75
6.5 Results and Discussion 76
6.5.1 Pre-Processing Step 76
6.5.2 Segmentation 76
6.5.3 Performance Analysis 77
6.6 Conclusion and Future Work 83

7 Conclusion 84
7.1 Conclusion 85
7.2 Future Perspective 86
7.3 French Conclusion 88
7.4 Perspectives 89

List of Figures

1.1 Artificial Intelligence and its Domains 2

2.1 Complete Graph K_6 11
2.2 Graph G and its Complement H 11
2.3 Graph G and a sub-graph H, K 12
2.4 Bipartite Graph 12
2.5 A and B graphs representing isomorphism, and C as non-isomorphic graph. The three graphs has four nodes and four edges. 13
2.6 The Representation of 4 x 4 Image as graph 16
2.7 An Image with its filtering 17
2.8 Different 3D data representations, obtained from [GZWY20] 18

3.1 Classification of shape representation techniques referenced by [ZL04] 22
3.2 [ZdFFeR07] Classification for 3D shape descriptors 23
3.3 Alternative taxonomy for 3D descriptors depending on Input Data 3.3a, and the Application 3.3b. Both Images are obtained from [RBRY19] 25
3.4 Schematic view of the classification criteria for shape correspondence 26
3.5 Conceptual Framework [TV08] 28
3.6 Two images showing both the device and result of CT device. 29
3.7 Brain PET Scan 30
3.8 A patient with uncontrolled complex partial seizures undergoes a SPECT scan. The left side of the brain's temporal lobe has less blood flow than the right, indicating to the surgeon that a nonfunctioning portion of the brain is producing the seizures. 31
3.9 MRI scan Images 32
3.10 Process flow of a brain tumor detection system 32
3.11 Schematic flow diagram shows the steps involved in the segmentation process.[MHH+22] 35
3.12 Diagram of graph building for a simple 2-D image. (This figure is based on Boykov's ECCV 2006 tutorial. Tutorial) 36

4.1 Neighborhood definition 40
4.2 Gaussian Curvature 41
4.3 Cosine Similarity between two models. 43
4.4 Bipartite graph for finding the similarity measure between descriptor vectors. V_{M1} and V_{M2} are the two sets of interest points with their corresponding $featurevector$. d_{ij} is the weighted link between every two vertices with cosine similarity as a value. 43
4.5 Examples of four 3D models with the key points detected in black on each using the Fraction-method. 45
4.6 Examples of four 3D models with the key points detected in black on each using the Cluster method. 45

4.7 Two different poses were matched using the Brute-force algorithm for cats in the first row, and hands in the second row, with the different parameters for neighborhood calculation 47
4.8 Two different poses were matched using the Hungarian algorithm for cats in the first row, and hands in the second row, with the different parameters for neighborhood calculation, blue and green lines refer to bad and good matches respectively. 47
4.9 One-to-One Correspondence match between two meshes from each class, showing a number of key points, as well the correct matches retrieved by Brute-force, and Hungarian algorithms. 48
4.10 Cosine Distance Matrix for the used 100 Model Database classified into 9 classes 48
4.11 Retrieval results of different queries 50

5.1 Flow-Chart of proposed approach 54
5.2 Neighboring Construction 55
5.3 Descriptor Maps. Left to Right: Ground Truth, proposed approach by [ECESAB20], ours 56
5.4 Descriptor Maps: Faces Reactions 57
5.5 Descriptor Maps: Elephant Poses 57
5.6 Both Descriptor Map, along with their Feature Region Clustering for several models 59
5.7 [GF08] Segmentation Results 60
5.8 [Pap14] shape deformations used to produce object class members . . 61
5.9 [SP04] Facial expressions replicated from scans onto a digital character 61
5.10 Matched Correspondences 62
5.12 3D matching within the same classes 62
5.11 Articulated 3D shape matching 63
5.13 Classification 64

6.1 Tumor Types 68
6.2 Referenced Image from [CR10] 69
6.3 Number of papers on brain tumor segmentation approaches from 200 till 2018, obtained from [HM19] 70
6.4 Partially cropped image from [HM19] showing the surveys done from 2012 till 2019 with the number of methods used in each 71
6.5 A directed capacitated graph, with costs of edges, is reflected by their thickness. A similar graph-cut construction was first used in vision by Greig et al. [GPS89] for a binary image. restoration. 74
6.6 Alpha-Expansion Graph 75
6.7 Steps from: Original Image towards Segmentation 76
6.8 Initial Images with their greyscale filtered images 77
6.9 The full approach compared to the ground truth of eight samples . . 78
6.10 Numerical Results for ten samples taken randomly. 80

6.11 Images that failed in their full segmentation 81
6.12 Samples that all of size 512x512 from [Che17] 82
6.13 Samples that has different sizes from [Cha19] 82

List of Tables

1.1 Graph Applications 3

3.1 3D shape descriptors' comparison 24

4.1 Numerical results for keypoints extraction, two poses of two models are shown 46

5.1 Time required for computation of the clusters in seconds 59

Chapter 1

Introduction

"Perseverance is the hard work you do after you get tired of doing the hard work you already did."

– Newt Gingrich

1.1 Motivation 2
1.1.1 3D shape matching and retrieval 4
1.1.2 Medical Imaging 4
1.2 Conducted Research Questions 4
1.3 Organization of the thesis 5
1.4 French Introduction 6
1.4.1 Motivation 6
1.4.2 Correspondance et récupération de formes 3D 7
1.4.3 Imagerie médicale 7
1.4.4 Questions de recherche traitées 8

Summary

In this chapter, we introduce our motivation and goal behind this thesis, in addition to revealing the structure of this manuscript. This is followed by the french version of the introduction

1.1 Motivation

Computer vision is a branch of artificial intelligence (AI) that allows computers and systems to extract useful information from digital photos, videos, and other visual input, as well as to perform actions or make recommendations based on these data. If artificial intelligence allows computers to think, computer vision allows them to see, watch, and comprehend. In Figure 1.1, we show a brief domain for artificial intelligence.

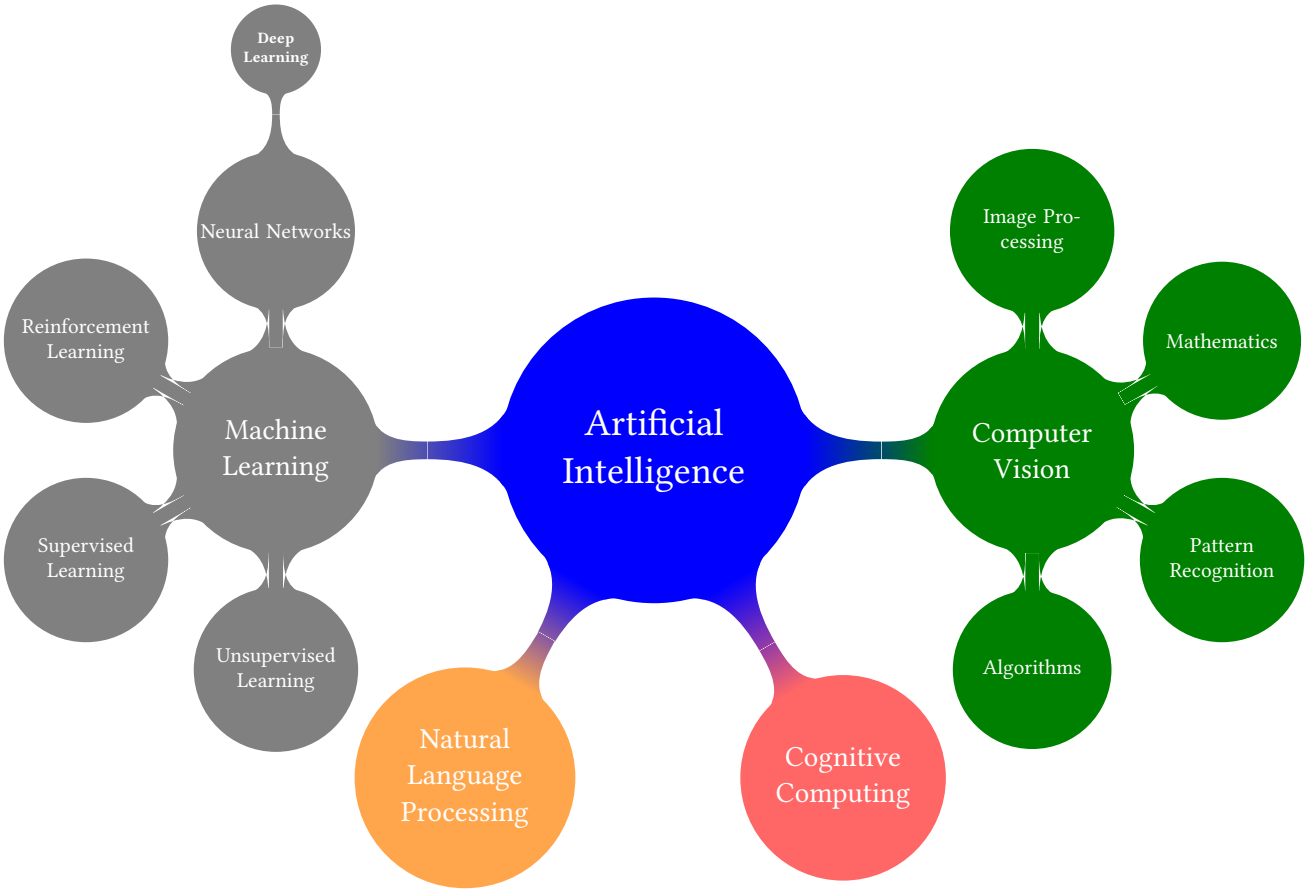


Figure 1.1: Artificial Intelligence and its Domains

At the same time, graph theory has been effectively used for a variety of problems in computer vision, ranging from low-level tasks (such as image segmentation, detection and tracking, stereo matching, and so on) to high-level tasks (such as image classification, object recognition, and image parsing). Graphs can also be used to rep-

resent a wide range of relationships and activities in physical, biological, social, and information systems. Table 1.1 shows some of the real-life applications that could be represented in graph theory.

Context	Graph Vertices and Edges
Biology	A graph is formed by the connections between atoms in molecules and crystal grids.
	A network can be used to model the transmission of diseases and epidemics.
Social Networks	Related persons connected as their relations between friends, relatives, etc...
Transportation Networks	Street ntersections connected by the highways
Image	Pixels connected by their relative intensities

Table 1.1: Graph Applications

When dealing with a group of objects or things, the first step toward comprehension is to organize them into classes or groups based on their commonalities. As a result, graphs are particularly appealing, as they allow for the modeling of relationships between various items. Graphs are the primary tool employed throughout the thesis.

In this thesis, we aimed to benefit from the well-representation of graphs in both three-dimensional (3D) and two-dimensional (2D) domains.

3D matching is a critical approach to detecting and aligning repeated or similar patterns in geometric objects. Shape comparison and retrieval; data fusion; pattern recognition; object classification; and data recovery are just a few of the data processing and analysis tasks that may help with computer graphics and vision. Despite its widespread use in computer graphics, vision, and medical image analysis, 3D matching computation is difficult when the matching data's corresponding regions or patterns lack saliency (due to small feature sizes or ambiguous characteristics) or have significant geometrical inconsistency (due to data losses, occlusion, or noise).

This was the first motivation for us to address it in the first section of our thesis. The second motivation for us was revealed by the belief that, nowadays, the medical field and life-saving are important. A brain tumor that considered one of the top three diseases that kills many people each year, this gives us the idea to deploy and work on the segmentation of brain tumors, which will lead to early diagnosis and, consequently, early treatment, wishing recovery.

More detailed information about the work conducted in this thesis is in the following two sections.

1.1.1 3D shape matching and retrieval

An automatic approach is required to assess the similarity between two items for the categorization and retrieval of 3D objects according to the content. The main premise of such an approach is that the measure of similarity between two 3D objects may be reduced to a distance computing between their descriptors.

It was our focus in this dissertation, where we study the features of the shapes, and how these features may benefit in finding the match between any two objects. To find a one-to-one correspondence between them, treating the model as an optimization problem, getting the optimal results using the Brute-Force, Hungarian, and other algorithms was a target.

The existing solutions for 3D shape matching and classification are quite robust concerning rigid transformations like rotation, translation, and even scale changes, but at the same time, these solutions lack shape-preserving transformations like isometric transformation.

In our contributions, we seek to treat this problem and use descriptors as well as matching algorithms to avoid un-preserving shape transformations.

1.1.2 Medical Imaging

A tumor is one of the most frequent brain diseases; according to World Health Organization (WHO) data, it accounts for one of the top ten causes of death globally. Accurate and timely diagnosis, as well as suitable treatment, can significantly reduce mortality.

Segmenting brain tumors is a difficult task since it typically comprises a huge amount of data with occasional artifacts due to patient movements, limited acquisition time, and soft tissue boundaries. There is also a huge group of tumor forms that come in a variety of shapes and sizes, can appear anywhere, and have varying image intensities. Some of them may also cause deformations in the surrounding structures of the brain. There are several MR acquisition procedures, each of which can provide different information about the brain.

Due to this fact, and since we can start treatment as soon as we diagnose the tumor, in our second part, we aim to design a system for segmenting a broad class of brain tumors in MR images reliably and accurately.

1.2 Conducted Research Questions

For this manuscript, we tend to answer the following questions for three-dimensional models:

1. What is the proper representation for 3D objects?

2. Which algorithm could be more efficient at matching two models?
3. How does clustering approach faster?
4. The importance of one-to-one correspondence.
5. How do the descriptors affect the matching phase and, consequently, the classification and retrieval phases?

On the other hand, this thesis answers the following questions related to the part of the brain tumor in the medical imaging field:

1. Because most tumors have heterogeneous appearances and manual segmentation is time-consuming, what method can be optimal in time and retrieval results to segment the tumor?
2. How can we properly classify tumor grade, which reflects treatment type?

1.3 Organization of the thesis

This dissertation is organized as follows:

We started in Chapter 2 by providing some preliminaries and explaining some notations that are necessary for the remainder of the thesis.

Since our thesis is organized into two distinct topics, we introduce in Chapter 3 the related work for 3D objects, the algorithms for matching and classification, in addition to the detectors and descriptors used before. At the same time, related work for the domain of brain tumor segmentation, in general, provides more details on the graph-based approaches that we follow for our framework.

In Chapter 4, we proposed a framework for finding a superior one-to-one correspondence between 3D models to obtain optimal matching and retrieval. To do so, we first detect feature points using the well-known 3D Harris Detector, followed by proposing a combination of local shape descriptors to form a compact feature vector for the key points extracted that consists of Gaussian Curvature, Curvature Index, and Shape Index. Lastly, we model the matching problem as a combinatorial problem solved using the Brute-Force approach and the Hungarian one, comparing the efficiency between them.

While in Chapter 5 we extend the novel compact feature vector, combining several more geometric representative curvatures, it is simple in complexity computation and powerful in the sense of affine transformations. The 3D surface is modeled as an undirected weighted graph, with the Gaussian kernel as a weight function. Integer Linear Programming is used to segment our meshes into regions where we maximize the modularity between vertices. These regions are represented by a single node each, which represents a subgraph for each model. These sub-models were matched using a linear sum assignment problem.

Chapter 6 is related to the brain tumor segmentation field, where we use a well-known algorithm that shows effectiveness for segmenting an image to detect tumors from MRI (Magnetic Resonance Images). As a first step, preprocessing MRI was performed to prepare them as input for a graph-based approach. The technique is based on the Boykov-Kolmogorov algorithm and the alpha-expansion method. This methodology tends to represent the image pixels as graph nodes, computing the weights between edges as the difference between intensities. This problem is formulated as an energy minimization problem and solved to find a 0,1 label for the image. As the last step for our approach, we tend to further do a post-processing step to enhance the overall segmentation.

With a summary of the contributions and some recommendations for additional research, we bring the thesis to a close in the final chapter 7, which mainly describes one of the promising concepts that we hope to develop in subsequent research.

1.4 French Introduction

1.4.1 Motivation

La vision par ordinateur est une branche de l'Intelligence Artificielle (IA) qui permet aux ordinateurs et aux systèmes d'extraire des informations utiles à partir de photos numériques, de vidéos et d'autres entrées visuelles, ainsi que d'effectuer des actions ou de faire des recommandations basées sur ces données.

Si l'intelligence artificielle permet aux ordinateurs de penser, la vision par ordinateur leur permet de voir, de regarder et de comprendre. Dans la figure 1.1, nous montrons un bref domaine pour l'intelligence artificielle. En même temps, la théorie des graphes a été utilisée efficacement pour résoudre une variété de problèmes de vision par ordinateur, allant des tâches de bas niveau (la segmentation d'images, la détection et le suivi d'images, la correspondance stéréo, etc.) aux tâches de haut niveau (la classification d'images, la reconnaissance d'objets, l'analyse d'images).

Les graphiques peuvent également être utilisés pour représenter un large éventail de relations et d'activités dans les systèmes physiques, biologiques, sociaux et d'information. Le tableau 1.1, applications de la théorie des graphes, montre certaines applications réelles qui pourraient être représentées dans la théorie des graphes.

Lorsqu'il s'agit d'un groupe d'objets, la première étape vers la compréhension consiste à les organiser en classes, ou en groupes, en fonction de leurs points communs. Par conséquent, les graphiques sont particulièrement attrayants, car ils permettent de modéliser les relations entre divers éléments. Les graphiques sont le principal outil utilisé tout au long de la thèse. Dans cette thèse, nous avons cherché à tirer parti de la bonne représentation des graphes dans les domaines tridimensionnel (3D) et bidimensionnel (2D).

La correspondance 3D est une approche essentielle pour détecter et aligner des

motifs répétés ou similaires dans des objets géométriques. Comparaison et récupération de forme, la fusion des données, la reconnaissance de formes, la classification d'objets et la récupération de données ne sont que quelques tâches de traitement et d'analyse de données qui peuvent aider à l'infographie et à la vision. Malgré son utilisation répandue dans l'infographie, la vision et l'analyse d'images médicales, le calcul de correspondance 3D est difficile lorsque les régions ou les modèles correspondants des données correspondantes manquent de saillance (en raison de petites tailles d'entités ou de caractéristiques ambiguës) ou présentent une incohérence géométrique importante (en raison de pertes de données, occlusion ou bruit).

C'était la première motivation pour nous de l'aborder dans la première section de notre thèse. La deuxième motivation pour nous a été révélée par la conviction que, de nos jours, le domaine médical et le sauvetage sont importants. Une tumeur au cerveau considérée comme l'une des trois principales maladies qui tue de nombreuses personnes chaque année, cela nous donne l'idée de déployer et de travailler sur la segmentation des tumeurs cérébrales, ce qui conduira à un diagnostic précoce et, par conséquent, un traitement précoce, souhaitant la guérison. Des informations plus détaillées sur les travaux menés dans cette thèse se trouvent dans les deux sections suivantes.

1.4.2 Correspondance et récupération de formes 3D

Une approche automatique est nécessaire pour évaluer la similarité entre deux éléments pour la catégorisation et la récupération des objets 3D en fonction du contenu. La prémisse principale d'une telle approche est que la mesure de similarité entre deux objets 3D peut être réduite à un calcul de distance entre leurs descripteurs.

C'était notre objectif dans cette thèse, où nous étudions les caractéristiques des formes et comment ces caractéristiques peuvent bénéficier de la recherche de la correspondance entre deux objets. Pour trouver une correspondance biunivoque entre eux, le traitement du modèle comme un problème d'optimisation et l'obtention des résultats optimaux à l'aide des différents algorithmes efficaces étaient un objectif. Les solutions existantes pour la correspondance et la classification des formes 3D sont assez robustes concernant les transformations rigides comme la rotation, la translation et même les changements d'échelle.

En même temps, ces solutions manquent de transformations préservant la forme comme la transformation isométrique. Dans nos contributions, nous cherchons à traiter ce problème et utilisons des descripteurs ainsi que des algorithmes d'appariement pour éviter les transformations de forme non conservatrices.

1.4.3 Imagerie médicale

Une tumeur est l'une des maladies cérébrales les plus fréquentes. Selon les données de l'Organisation Mondiale de la Santé (OMS), elle représente l'une des dix prin-

cipales causes de décès dans le monde. Un diagnostic précis et opportun, ainsi qu'un traitement approprié, peuvent réduire considérablement la mortalité.

La segmentation des tumeurs cérébrales est une tâche difficile car elle comprend généralement une énorme quantité de données avec des artefacts occasionnels dus aux mouvements du patient, au temps d'acquisition limité et aux limites des tissus mous.

Il existe également un grand nombre de formes de tumeurs qui se présentent sous différentes formes et tailles, peuvent apparaître n'importe où et avoir des intensités d'image variables. Certains d'entre eux peuvent également provoquer des déformations dans les structures environnantes du cerveau.

Il existe plusieurs procédures d'acquisition IRM, chacune pouvant fournir des informations différentes sur le cerveau. De ce fait, et puisque nous pouvons commencer le traitement dès que nous diagnostiquons la tumeur, dans notre deuxième partie, nous visons à concevoir un système permettant de segmenter une large classe de tumeurs cérébrales dans les images IRM de manière fiable et précise.

1.4.4 Questions de recherche traitées

Dans cette thèse, nous avons tendance à répondre aux questions suivantes pour les modèles tridimensionnels :

1. Quelle est la bonne représentation des objets 3D ?
2. Quel algorithme pourrait être le plus efficace pour faire correspondre deux modèles ?
3. Comment le regroupement s'approche-t-il plus rapidement ?
4. Quelle est l'importance de la correspondance un par un (one-to-one correspondence) ?
5. Comment les descripteurs affectent-ils la phase de correspondance et, par conséquent, les phases de classification et de récupération ?

D'autre part, cette thèse répond aux questions suivantes liées à la place de la tumeur cérébrale dans le domaine de l'imagerie médicale :

1. Étant donné que la plupart des tumeurs ont des apparences hétérogènes et que la segmentation manuelle prend du temps, quelle méthode peut être optimale en termes de temps et de résultats de récupération pour segmenter la tumeur ?
2. Comment pouvons-nous classer correctement le grade de la tumeur, qui reflète le type de traitement ?

Chapter 2

Preliminary Notions

"The beginning of knowledge is the discovery of something we do not understand."

– Frank Herbert

2.1 Graphs 10
2.1.1 Basic Definitions and Notations 10
2.1.2 Graph matching 13
2.2 Optimization 14
2.2.1 Continuous optimization 15
2.2.2 Discrete optimization 15
2.2.3 Detailed Discrete optimization 15
2.3 Two and Three-Dimensional Models 17

Summary

This chapter is devoted to defining some terminologies that will be used through out this thesis. All related terms, notations are settled in this chapter, specific terminologies will be added later for chapters convenient.

2.1 Graphs

In graph theory, a graph is a representation of a data set where some pairs of data items are connected by links [Gal13, VL07]. The data elements are called vertices or nodes and the connections are called edges or arcs. In most contexts, a graph is represented as G = (V, E), where V is a set of vertices and E is a set of edges between vertices.

2.1.1 Basic Definitions and Notations

We use standard graph-theoretical terminology, and we assume that the reader is familiar with the basic concepts of graph theory. The notation of this thesis is mainly conventional and can be found, for example, in [AS98, BM08].

Each graph can be viewed as weighted. Taking the notation above, the vertex weight is a function ω : V$\rightarrow \mathbb{R}$ and the edge weight is a function ω : E$\rightarrow \mathbb{R}$. To simplify the notation, we use ω to refer to both the vertex and the obtain edge weighting. The weight of a vertex is denoted by ω_i, and the weight of an incident edge on two vertices is denoted by ω_{ij}.

If ω_i = 1, $\forall \nu_i \subset$ V and ω_{ij} = 1, $\forall e_{ij} \subset E$, then we can consider the graph to be unweighted. Unless otherwise specified, all node and edge weights are considered equal to one.

We treat ω_{ij} = 0 as equivalent to $e_{ij} \notin E$. Intuitively, a zero edge weight means that the edge is not a member of the edge set. Each edge is considered oriented and some edges are additionally directed. An orientation of an edge means that every edge $e_{ij} \in E$ contains an order of vertices v_i and v_j. An edge e_{ij} is directed if ω_{ij}= ω_{ji}.

A graph in which none of the edges is directed is called a **undirected graph**, and a graph in which at least one edge is directed is called a **directed graph** or **digraph**. A directed edge is represented by the notation $e_{i\rightarrow j}$.

In this dissertation our mainly focus is on the undirected weighted graphs.

Undirected graph G = (V, E) is:

- connected if, between each pair of vertices, there is a path

- acyclic if it has no cycle
- a tree if it is connected and acyclic
- bipartite if V can be partitioned into (at most) two independent sets

Definition 2.1.1 (Complete Graph)**.** is a basic undirected graph with a single edge connecting each pair of different vertices, and it is denoted by K_n. Maximal cliques occur across all complete graphs. They are maximally connected as the whole set of vertices. An empty graph is the complement graph of a complete graph.

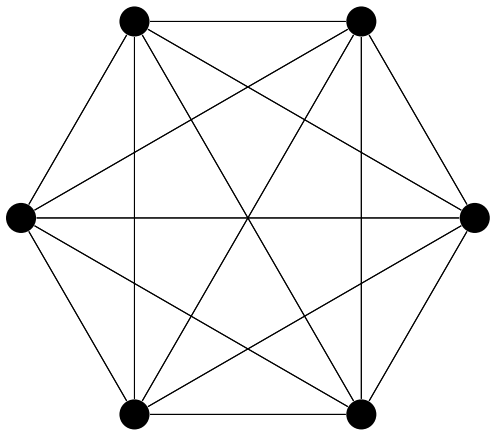
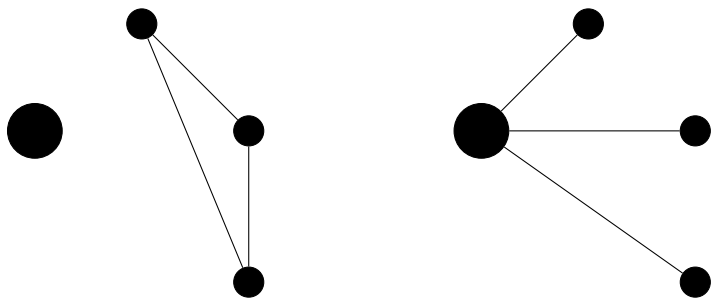


Figure 2.1: Complete Graph K_6

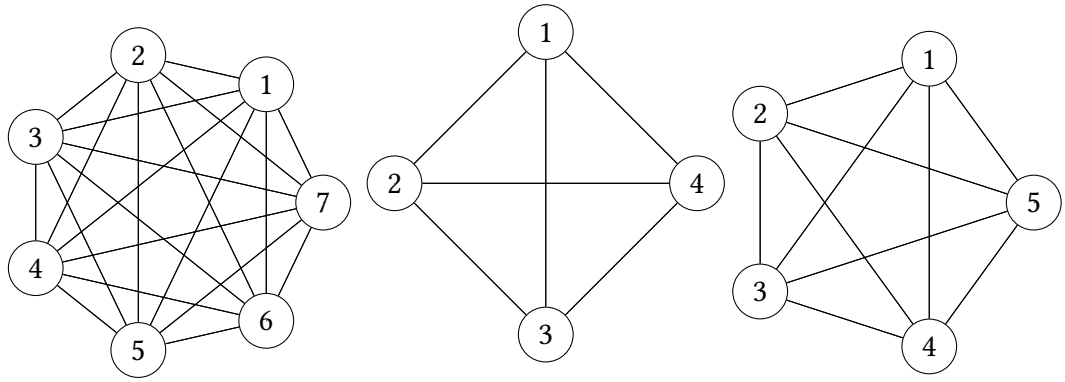
Definition 2.1.2 (Complement Graph)**.** or named as inverse graph is the graph on the same vertices of G where the adjacent vertices in H are not adjacent in G.



(a) Graph G with four nodes (b) Graph H, the complement of G

Figure 2.2: Graph G and its Complement H

Definition 2.1.3 (Sub-graph)**.** A graph H is called a sub-graph of a graph G, written H $\subseteq$ G, if V(H) $\subseteq$ V(G) and E(H) $\subseteq$ E(G). We also say that G contains H as a sub-graph. If H $\subseteq$ G and either V(H) is a proper subset of V(G) or E(H) is a proper subset of E(G), then H is a proper sub-graph of G. If a sub-graph of a graph G has the same set of vertex as G, then it is a subgraph that stretches G.



(a) Graph G with seven nodes (b) A sub-graph H (c) A sub-graph K

Figure 2.3: Graph G and a sub-graph H, K

Definition 2.1.4 (Bipartite Graph)**.** A graph is called a bipartite graph if V can be partitioned into two subsets V1 $\subset$ V and V2 $\subset$ V , where V1$\cap$ V2 = ϕ and V1 $\cup$ V2 = V , such that E $\subseteq$ V1 $\times$ V2.

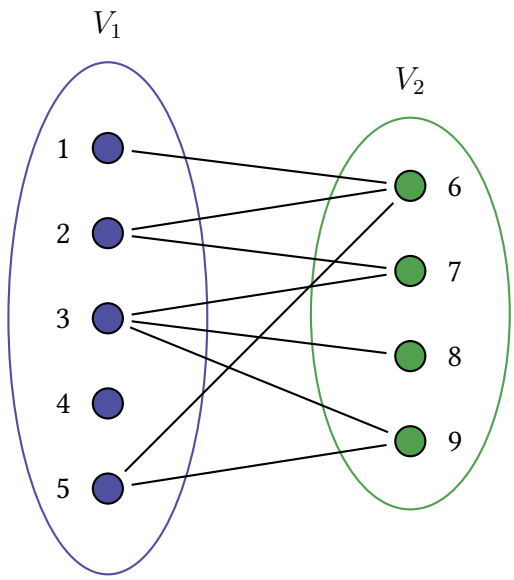


Figure 2.4: Bipartite Graph

Definition 2.1.5 (Degree of the Vertex)**.** The degree of a vertex v in a graph G is the number of edges incident with v and is denoted by deg_Gv or simply by deg v if the graph G is clear from the context. The set $N(v)$ of neighbors of a vertex v is called the neighborhood of v. Thus deg v=$|N(v)|$. A vertex of degree 0 is referred to as an isolated vertex, and a vertex of degree 1 is an end-vertex(or a leaf). The minimum degree of G is the minimum degree among the vertices of G and is denoted by $\delta(G)$, the maximum degree of G is denoted by $\Delta(G)$.

Definition 2.1.6 (Graph Partitioning)**.** A graph partition is the reduction of a graph to a smaller graph with the aid of partitioning its set of nodes into jointly extraordinary groups.

Edges of the unique graph that move between the groups will produce edges within side the partitioned graph. If the wide variety of resulting edges is small in comparison to the original graph, then the partitioned graph can be better proper for evaluation and problem-fixing than the original.

2.1.2 Graph matching

Graph matching is the process of identifying a relationship between two graphs' nodes and edges. The matching is either exact or in-exact, where the first one is about preserving the edges while mapping vertices; this means that if two vertices are adjacent in graph A, when mapping to graph B, they shall be adjacent as well. And if this relationship is bijective, we get the well-known graph isomorphism problem.

While in-exact graph matching corresponds to matching problems in which exact matching is unattainable, for as when the number of vertices in the two graphs differs. In this scenario, the best possible match is necessary.

In image recognition applications, for example, the findings of image segmentation in image processing frequently create data graphs with many more vertices than the model graphs data is anticipated to match against. Even though the number of vertices and edges in an attributed graph is the same, the matching may be merely approximate [Ben02].

The most extreme type in Exact Graph matching is **isomorphism**, where each node in the first graph must have a one-to-one correlation with each node in the second graph. This type is our main focus in this research due to the several application applied in this domain.

Graph isomorphism is a significant concept for determining if two items are the same, subject to the invariance properties of the underlying graph representation [Bun00].

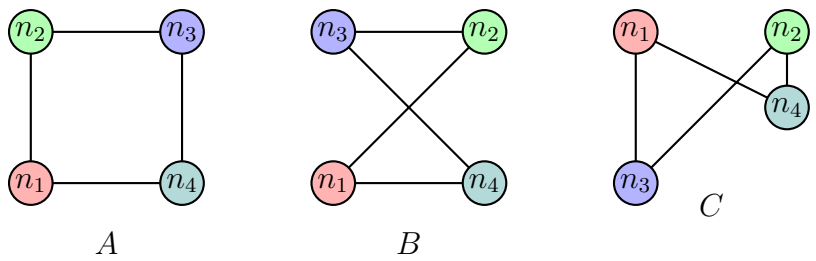


Figure 2.5: A and B graphs representing isomorphism, and C as non-isomorphic graph. The three graphs has four nodes and four edges.

Graphs might appear to be distinct, but they are all the same. It is only important to know which nodes are connected to which other nodes. The two graphs are said to be isomorphic if there is an edge-preserving bijection between them, that is if there is a function that maps nodes from one graph onto those of another while keeping the set of connections for each node the same.

This is shown in the first two graphs in figure 2.5, where despite the visual shape, they are isomorphism, at the same time, graph C which has the same number of nodes and edges, can not be isomorphic for them due to the change in the connected edges.

2.2 Optimization

Optimization, known as mathematical optimization or programming, is the process of selecting the optimal element from a group of possibilities based on some criterion. It is used to solve quantitative issues in a variety of fields, including physics, biology, engineering, economics, and business.

Depending on whether the variables are continuous or discrete, optimization problems can be categorized into two groups:

1. **Continuous optimization** is a problem with continuous variables in which an optimal value from a continuous function must be determined.
2. **Discrete optimization** is an optimization problem with discrete variables in which an object such as an integer, permutation, or graph must be found from a countable collection.

Main terms used in any optimization modelling:

Definition 2.2.1 (Objective Function)**.** It is a real-valued function whose value is to be minimized or maximized over a range of possible options, for example $f(x) = x^2 + x + c$.

Definition 2.2.2 (Variables, or Decision variables)**.** Are variables whose values can change over a range of possible options in order to increase or decrease the objective function's value. In the example above it refers to the set of x values

Definition 2.2.3 (Feasible Region, or Constraints)**.** It is the whole set of items for the decision variables that perhaps the objective function should be optimized throughout.

Assuming such a problem:

$$\min f(x) \text{ subject to } x \subset K \ ,$$

where $f : \mathbb{R}^{\ltimes} \to \mathbb{R}$ is the objective function and $K \subset \mathbb{R}^{\ltimes}$ is the set of feasible set. Depending on the set K two categories are differentiated:

2.2.1 Continuous optimization

K is an uncountable infinite feasible collection. Interestingly, this may pay off: linear programming is polynomial solved, but it is NP-hard due to the additional functionality condition. Linear programming (LP) and different kinds of nonlinear programming are common problems (such as semi-definite programming, convex programming or quadratic programming).

2.2.2 Discrete optimization

The set K is (generally) finite, but it is usually large enough to examine and process all feasible solutions. The shortest path problem, minimum spanning tree problem, and minimum matching problem in a graph all examples.

In contrast, other problems in discrete optimization like Integer linear programming, the knapsack problem, finding the max-cut of a graph and the travelling salesman problem, are NP-hard problems.

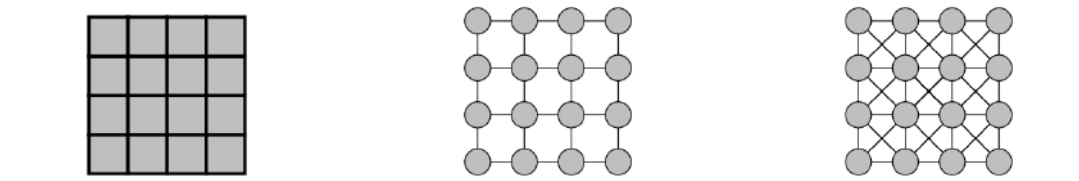
In this manuscript all the optimization problems we deal with, are discrete, more about integer linear programming and graph cut will be detailed in the following sections.

2.2.3 Detailed Discrete optimization

Image segmentation, which seeks to divide an image (set of pixels) into several objects (subsets of pixels) sharing the same features, is a classic problem in computer vision. To put it another way, image segmentation seeks to give each pixel in a picture a label. The same label is given to pixels that have the same intensity, color, texture, etc.

There are numerous methods that fall under different frameworks, including thresholding and clustering, variational techniques, and graph partitioning techniques, to mention a few. The term "graph partitioning methods" refers to a variety of techniques, including, for example, normalized cuts, random walking, least spanning tree, and minimum cut. The same concepts are applied to the three-dimensional models. For simplicity we will talk about images for instance.

An image can be described as a structured graph in which the image's pixels are connected to the nodes. A node's four closest neighbors, or its four "nearest" pixels in the cardinal directions, are traditionally connected to it by edges. It is possible to connect with more "nearest" neighbors using a variety of more complicated graphs, such as an eight nearest neighbors structure. Possible graph constructions from an image are shown in Fig. 2.6.



(a) Image with 4 × 4 pixels. (b) 4-connected pixel graph. (c) 8-connected pixel graph.

Figure 2.6: The Representation of 4 x 4 Image as graph

Integer Linear Programming

Abbreviared as ILP.

ILP in standard form is expressed as

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}^{\mathrm{T}}\mathbf{x} \\ \text{subject to} \quad & A\mathbf{x} + \mathbf{s} = \mathbf{b}, \\ & \mathbf{s} \geq \mathbf{0}, \\ & \mathbf{x} \geq \mathbf{0}, \\ \text{and} \quad & \mathbf{x} \in \mathbb{Z}^n, \end{aligned}$$

where $\mathbf{c}, \mathbf{b}$ are vectors and A is a matrix, where all entries are integers.

As with linear programs, ILPs not in standard form can be converted to standard form by eliminating inequalities, introducing slack variables (s) and replacing variables that are not sign-constrained with the difference of two sign-constrained variables

Graph Cut

A cut $C = (S, T)$ is a partition of V of a graph $G = (V, E)$ into two subsets S and T. The cut-set of a cut $C = (S, T)$ is the set $\{(u, v) \in E \mid u \in S, v \in T\}$ of edges that have one endpoint in S and the other endpoint in T. An s-t cut is a cut in which s belongs to the set S and t belongs to the set T if s and t are specified vertices of the graph G.

The quantity of edges that traverse a cut determines its size or weight in an unweighted undirected graph. The value or weight of a weighted graph is determined by adding the weights of the edges that cross the cut. A cut-set that lacks any other cut-set as a suitable subset is said to be a bond.

Image Segmentation

- Image: $x \in \{R, G, B\}^N$
- Output: Segmentation (also called opacity) $S \in R^N$ (soft segmentation). For

hard segmentation $S \in \{0 \text{ for background}, 1 \text{ for foreground/object to be detected}\}^N$

- Energy function: $E(x, S, C, \lambda)$ where C is the color parameter and λ is the coherence parameter. $E(x, S, C, \lambda) = E_{\text{color}} + E_{\text{coherence}}$
- Optimization: The segmentation can be estimated as a global minimum over S: $\arg\min_S E(x, S, C, \lambda)$

2.3 Two and Three-Dimensional Models

In our domain of work, when talking about two-dimensional model, we are relating it to the digital images.

An image is a function of intensity values over a 2D plane, the sample points in an image is its pixels. Image could be represented in several forms, like the colored image: three color planes each having 8 bits, for example the RGB one. There is the gray-scale image that has a single color plane with 8 bits, in addition to a black-white image which has 2 bits single plane. Figure below shows and example for an image with a processing function applied to it: Filtering;

Figure 2.7: An Image with its filtering

Images are almost everywhere, and found in different fields of computer or machine vision, and medical analysis. It is widely spread in the concept of patter recognition, motion detection, segmentation, matching, etc...

In the other part, we have the three dimensional models, 3D models could be found in several representations, as surfaces; whether it is presented by mesh, parametric, or as a raw data; it could be as point cloud or voxels. 3D models are to limited to these.

Figure 2.8 shows different examples for 3D models related to several fields.

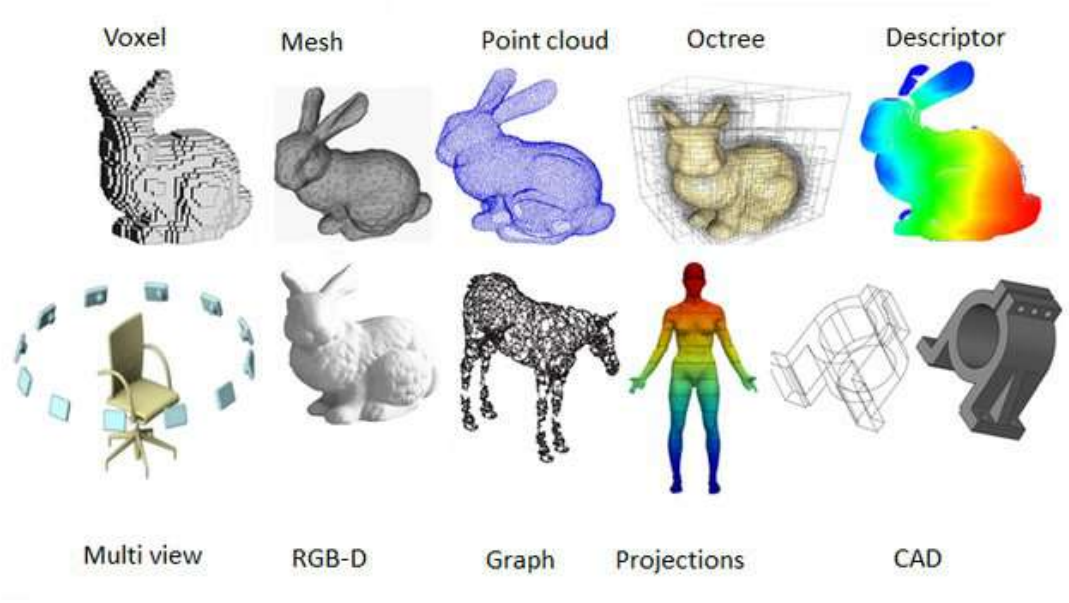


Figure 2.8: Different 3D data representations, obtained from [GZWY20]

Detectors

With the intention of using less data for complex vision tasks, the interest point detection issue originated in the computer vision field.

For instance, an interest point detector in a human-shaped model should choose vertices on the face, hands, and feet. Second, 3D mesh topology is arbitrary. In other words, a vertex can have any number of vertices that are close to it. As a result, choosing a vertex's neighborhood is more difficult.

Additionally, this flaw makes it possible for many tessellations to depict the same locality; as a result, an interest point detector should be able to handle it. Third, it is uncertain or difficult to compute the size of a locality in which a vertex is an interest point without a clear topological structure for meshes. Last but not least, the vertices' locations and their connectedness are the only pieces of information available. This fact makes the process more difficult because the level of interest must be determined using the information at hand, which also depends on the mesh topology.

In a different situation, image interest point detectors have improved to a usable level of efficacy. The well-defined image structure and interest spots that coincide with pixels that reflect intriguing structures in the scene that was caught by the image account for this.

Descriptors

A shape descriptor is a brief but informative description that identifies a 3D object as belonging to a certain category. Also it can be seen as an abstraction of the 3D

model that captures key shape details in an easy-to-compare structure.

Many shape matching applications use a shape descriptor to express a three-dimensional (3D) model using a fixed-dimensional vector, which reduces model comparison to computing the distance between two points in Euclidean space. Since it is simple to calculate the distance between two sites, the underlying matching is effective and the real-time demands of a retrieval system can be met.

Chapter 3

State of the art

"There is a great deal of difference between an eager man who wants to read a book and the tired man who wants a book to read."

– GK Chesterton

3.1 3D objects matching and classification 21
3.1.1 3D Objects Detectors and Descriptors 21
3.1.2 3D Objects Matching, and Classification 26
3.2 Brain Tumor Segmentation 28
3.2.1 Brain Imagining Modality 29
3.2.2 Approaches for Brain Tumor Segmentation 32

Summary

This chapter is devoted to the related work of our research. We divide the presentation of the literature according to the two main subjects of this thesis. First the related part to the 3D objects matching and classification, and the second one for the 2D brain tumor segmentation.

3.1 3D objects matching and classification

The main idea that facilitates the matching between objects and leads later to classification is the good representation of the 3D object, in addition to finding the appropriate similarity function to measure distances between entities in the feature space.

3.1.1 3D Objects Detectors and Descriptors

Feature detection is the task of observing interest points from the whole shape that is well remarkable and repeatable despite the variations that happen to the shape, and on the other side, feature description is the representation of the shape properties at the detected points or all points when working as a dense descriptor.

[DCG12] made benchmark consisted of 43 triangular mesh of hand-marked interest points, they evaluated different detection algorithms like Mesh Saliency, Scale Dependent corners (SD-corners), 3D-Harris, 3D-SIFT, and a multiscale detector based on Heat Kernel Signature (HKS). The results show that the 3D Harris detector obtains a good number of interest points more than in HKS and less than Mesh Saliency and SD-corners, and these points obtained were compared to that annotated by humans.

By referring to the main study of the 3D Harris detector in [SB11] it shows that the interest points extracted are repeatable and meaningful, especially for the usage of the similarity measure, and it is fast enough. Due to the studies that show the effectiveness of 3D Harris, this encourages using it to lower our data-bound, instead of working on a full model, consisting of 3000 vertex, for example, we take into consideration only 10 % of these vertices.

Once the interest points are extracted, or even working on the full model, shape descriptors are defined from these points using some data structure representation, several exciting surveys were done on 2D and 3D shape descriptors but not limited to [TG12, WOBL17, ZL04, ZdFFeR07].

[MP07] had evaluated the performance of different well-known detectors and descriptors based on 3D object matching across viewpoints and lighting conditions. Their study shows that although an invariant detector provides higher repeatability for large affine distortions, it is costly in computing. And this is a trade-off that in our approach we tried to optimize.

[ZL04] had classified the shape representation and description techniques as shown in the figure below.

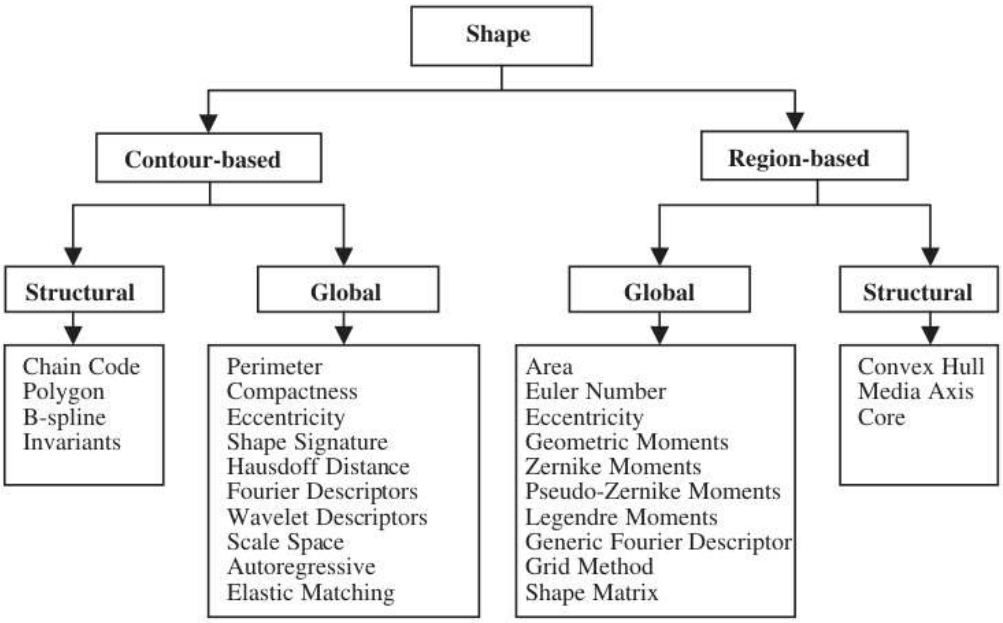


Figure 3.1: Classification of shape representation techniques referenced by [ZL04]

Their classification is depending on whether the features extracted are obtained from the contours of the shape only or the whole shape. In Structural based methods, the shapes are divided into segments called primitives, while for the global-based methods, usually there is a multi-dimensional computation for the feature vector.

Another study [ZdFFeR07], shows that the 3D shape descriptors are classified into feature-based, graph-based, and others not related to any one of the first two. Each of these categories has several descriptors' types underlying in the big title, but the main thing is that they are not disjoint, in another word, for example, we have distribution-based descriptors lying under the feature-based, it describes the global features in some way, such as shape distribution descriptor. Another example, a graph-based shape descriptor can also be viewed as a global feature-based descriptor.

Figure 3.2 shows the detailed classification for the 3D shape descriptors. In their survey, they show that the shapes are esteemed by measuring and comparing the features.

A descriptor is set to be good if it is compact, robust, and descriptive [GBS+16]. At the same time, having such a descriptor will help in retrieving, classifying, matching, and clustering between models. The main challenge in 3D object applications is to obtain a robust descriptor by extracting geometric and topological features, creating a special significance for the model to distinguish it.

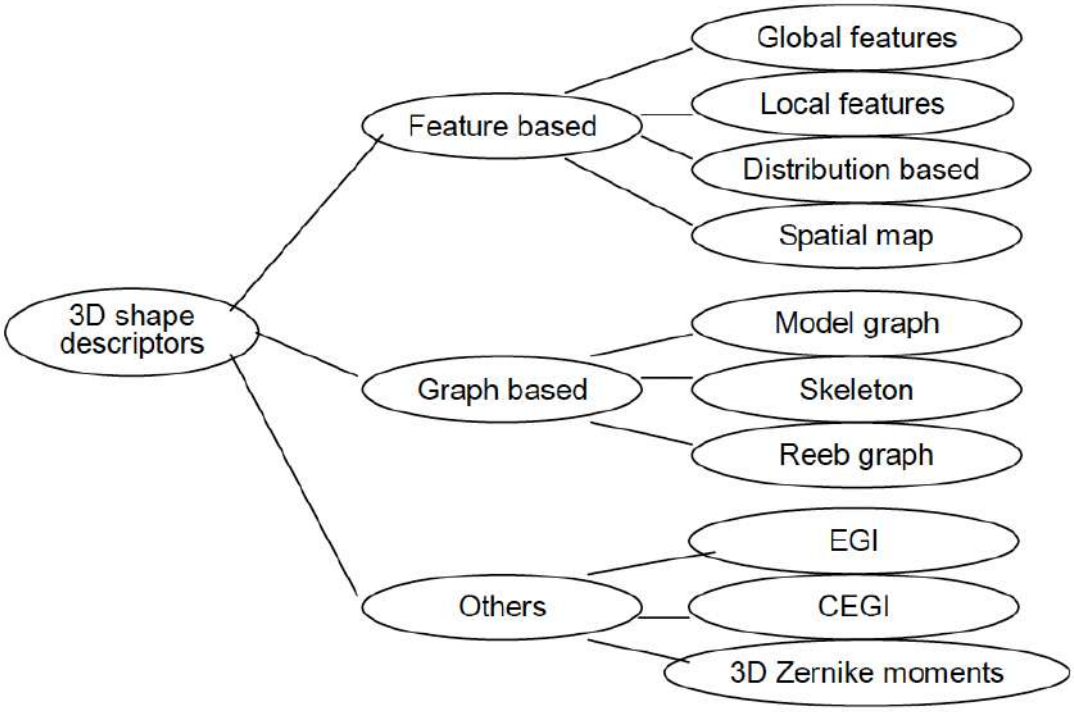


Figure 3.2: [ZdFFeR07] Classification for 3D shape descriptors

[LNJ+17] had discriminated between two main types of descriptors, global and local ones.

- Global features: Most often the descriptors in this category are used with other methods, where they are considered as an active filter and do not discriminate in the local parts of the model.
- Local features: The main importance of these descriptors is that it allows a perception of complex objects. It works simply by obtaining features on specific points on the shape, with no need for the whole object. Local features offer a unique measure of similarity for each object.

[BA16, HPPLG11, HPPL+12, TG12, WOBL17], according to these surveys, local features shows more effectiveness than the global ones. Due to the high complexity of graph-based features [HSKK01, SSGD03] that we do not seek in our approaches, we limited our search in the local and global based features. We represent in Table 3.1 a basic taxonomy relying on the shape information obtained and used in matching with the references of their study.

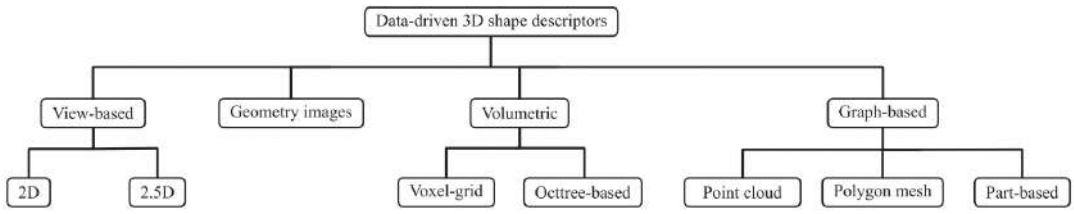
Category of method	Descriptor	Shape Feature	Limitations	Ref.
Local-based features	Shape spectrum	The distribution of the shape index over the entire mesh	a. Rough 3D data must be converted into useable geometrical surfaces; b. It is desirable to integrate with some global representation methods.	[Koe90]
	3D shape contexts	A coarse histogram of the relative coordinates of the remaining surface points	a. It's inefficient; b. It's difficult to index; c. The produced dissimilarity measure doesn't follow the triangle inequality.	[KPNK03]
	Point Feature Histogram(PFH)	Encodes a point's k-neighborhood geometrical properties by generalizing the mean curvature around the point using a multi-dimensional histogram of values.	It has a complexity of O(n.k^2) for a point cloud with n points.	[RBMB08]
	Spin Image	It is obtained by its own coordinates and the surface normal.	In case of high-level noise there will be a degradation in the performance.	[JH98]
Global-based features	Clustered Viewpoint Feature Histogram (CVFH)	Applies a region growing algorithm after removing the points with high curvature.	It lacks the notion of an aligned Euclidean space which causes the feature to miss a proper spatial description.	[AVB+11]
	Shape Distribution on Voxel Surfaces (SDVS)	Depends on the created voxel grid created on the surface.	Cannot address the confusion shape well.	[WV11]
	Cord and angle histograms	Histograms of the length and the angles of the cord rays	Not well discriminating for object details.	[PRM+00, AKKS99, PR99]

Table 3.1: 3D shape descriptors' comparison

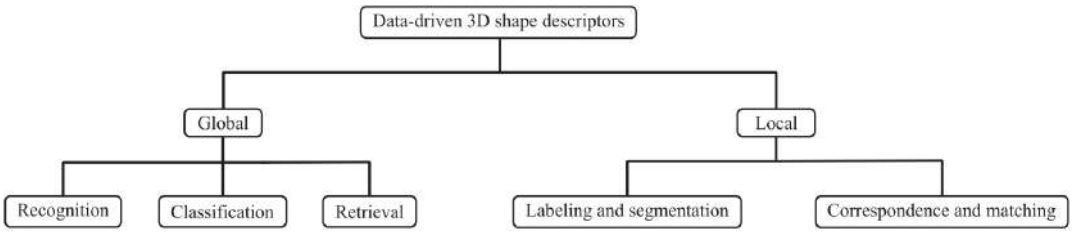
The classification of 3D descriptors according to [RBRY19] had relying on the type of the input, whether it is mesh, point cloud, depth images, volumetric, etc... One reason behind such categorisation is the fact that applying an approach for 3D mesh, would not be straightforward applied on RGB images due to the unstructured/ structured nature of the input.

At the same time, and in the same study, they showed that the descriptors can be divided depending on the type of the application to be used with, where this classifications assist and facilitate the choose of the descriptor, and this was shown previously in the study of [LNJ+17], where this categorization is only into global and local, and whether the application is related to classification, retrieval, correspondence, etc..., someone can choose the proper descriptor.

Figure 3.3 shows the classification for the study obtained by [RBRY19]. Due to



(a) Considering Input Data



(b) Considering the Application

Figure 3.3: Alternative taxonomy for 3D descriptors depending on Input Data 3.3a, and the Application 3.3b. Both Images are obtained from [RBRY19]

the several studies showing the effectiveness of local shape features, we tend to use them in our approaches.

Local shape descriptors depending on the curvatures like, Gaussian curvature, shape index [KVD92], mean curvature, normal distribution, etc... gives discriminating results [HPPLG11]. And depending on the work of Sarah Tang and Afzal Godil in [TG12] that shows open research in the combination of several local descriptors, we propose the combination of such well-effective descriptors in both Chapters 4,5.

In addition, the study of [RBRY19] it is obviously that due to the type of our input: meshes, and our application: correspondence and matching, our suggestions

to be used are accurate. We tend to use graph based as well local descriptors.

The main challenge regarding shape descriptors or detectors is to find such one with efficient representation, can be robust to several affine transformations, in addition, to be useful for partial matching in cases that full objects are not available.

3.1.2 3D Objects Matching, and Classification

Feature matching is the task where there is an establishment for a set of feature correspondences between the two sets of descriptors, and this matching is used in several applications, like shape correspondence, shape retrieval, recognition, classification, etc..

Upon the feature extraction, we will have the ability to match objects. Finding the correspondence can be obtained directly from the similarity between objects, or by aligning them first, then deriving a correspondence from the proximity of the aligned elements. A schematic view obtained from [VKZHCO11] is shown in the figure below. [KXX19] had proposed an improved version of the curve evolution

Input	**Geometry representation**		Points, feature points, surfaces, skeletons
	Dimensionality of the data		2D, 3D, 2D+time, 3D+time
Output	**Correspondence representation**	Correspondence + transformation	Translation, rigid, similarity, affine, nonlinear
		Correspondence only	Bijective, injective, many-to-many, probabilistic, crisp
	Full vs. partial		
	Dense vs. sparse		
Objective function	**Similarity-based correspondence**		Similarity only, similarity + distortion
	Rigid alignment		Largest common pointset, geometric distance
	Non-rigid alignment		Geometric distance + regularization
Approach	**Solution paradigm**	Transformation search	Alignment, pose clustering, non-rigid alignment
		Correspondence search	Continuous optimization, combinatorial search
		Hybrid search	ICP, prealignment + ICP, embedding + ICP
	Fully-automatic vs. semi-automatic		
	Global vs. local search		
	Pairwise vs. groupwise		

Figure 3.4: Schematic view of the classification criteria for shape correspondence

algorithm to describe the shape contour in a better way, at the same time they apply the cyclic Smith-Waterman algorithm for the matching process.

[YH09] employed normalized cross-correlation (NCC) as the similarity measurement and got good anti-noise results, however, NCC is time-consuming to compute. Only features of the same length can be handled by Euclidean distance and NCC.

Euclidean distance and NCC appear powerless in the presence of deformation or occlusion. Edit distance (ED) is initially applied to string matching by determining the smallest number of operations required to equalize two sequences using substitution, insertion, and deletion operations. ED can handle the matching of sequences of various lengths by using a threshold to assess whether the two features are equal. However, ED just counts the bare minimum of operations; once the difference between the two characteristics is within the threshold, it is ignored.

Lian Z. et al. in [LGB+13] had created a benchmark of 600 watertight triangle meshes that are derived from 30 original models, among which 26 objects are collected from several freely-accessible repositories (e.g., McGill database [SZM+08], PSB database [SMKF04], TOSCA shapes [BBO12], etc.). Then in their work and to match any two objects, they compute the dissimilarity measure generating a distance matrix, by analyzing these distance matrices they evaluate the performance of the retrieval system.

They apply eleven methods of descriptors and compared them using the top five quantitative measures (Nearest Neighbor, Precision-recall curve, E-measure, First Tier (FT) and Second Tier (ST), Discounted Cumulative Gain).

Matching objects lead to the ability to classify and even retrieve the best matches from a data set. According to [BS12] after representing a shape as a set of features, the matching step could be seen as an optimization problem to find the optimal transformation function, a similarity measure is defined, and by maximizing this similarity, the best one-to-one correspondence is found.

A dissimilarity measure must be used to calculate the distances between pairs of descriptors to measure the shape similarity of two objects. Although the term "similarity" is widely used, "dissimilarity" is more closely related to the concept of "distance": A little distance indicates a modest dissimilarity, while a large distance indicates a large similarity [TV08].

Several measures were proposed, for specific:

- Mahalanobis distance: the distance measure that estimates the similarity between two random multidimensional points;
- Euclidean distance: the ordinary measure between points in the Cartesian coordinate system of 2D or 3D spaces;
- Hausdorff distance: based on two sets of points of varying sizes with no one-to-one correlation between all points;
- Cosine distance: The similarity of two vectors in an inner product space is evaluated. It checks whether two vectors are pointing in the same direction by measuring the cosine of the angle between them.

One of the most used frameworks in the domain of shape retrieval and classification

is shown in Figure 3.5, where the first part needed is to extract the features from the model (descriptors), and with having a wide range of descriptors stored, whenever a query is introduced, descriptors are extracted and they are matched with optimal and more similar objects, and when such matching found the results are retrieved and visualized.

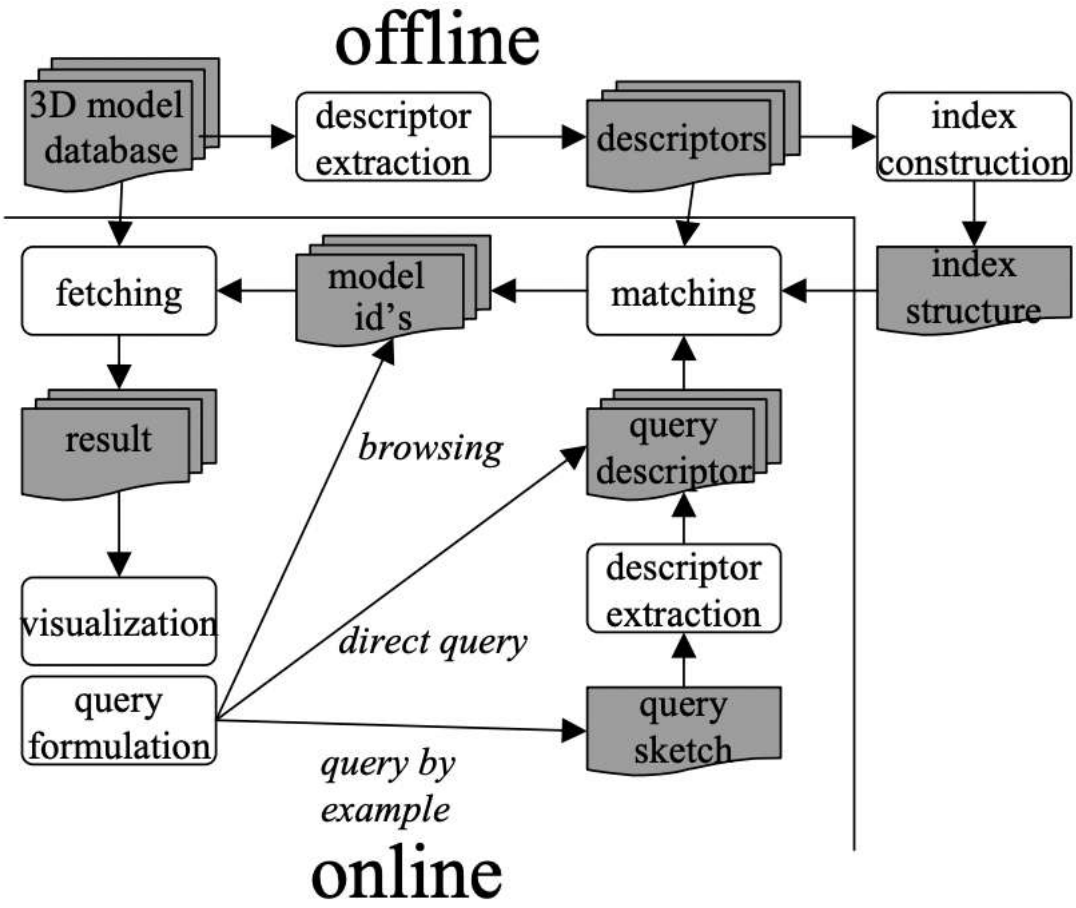


Figure 3.5: Conceptual Framework [TV08]

3.2 Brain Tumor Segmentation

Cancer is one of the most fatal diseases worldwide spread. It is affecting all ages [ZTC+20, MEJ20]. All brain cancers are tumors, but not all brain tumors are cancerous. Noncancerous brain tumors are called benign brain tumors. In this study, we seek to make a segmentation for the brain tumor because manual segmentation is a difficult and time-consuming task. In addition, it is difficult to understand the clinical manifestation of a brain tumor due to the variety in size, localization, the pace of growth, and pathology.

A brain tumor, on the other hand, is an abnormal mass of tissue in which some cells proliferate and reproduce out of control. This excessive growth takes up space inside the skull, disrupting normal brain activity and causing harm to brain cells. Increased pressure in the brain, displacing the brain or pressing against the skull, and invading nerves and healthy brain tissues are all possible causes of harm.

We seek to find an automatic segmentation that helps in the early detection of tumors, and subsequently early diagnosis and treatment. Many studies show how a tumor could be detected through graph-based, deep-learning, threshold-based, unsupervised methods, etc...

But mainly, we will introduce first the wide variety of the imaging modalities used for the brain. Computed tomography (CT), positron emission tomography (PET), single-photon emission computed tomography (SPECT), and Magnetic Resonance Imaging (MRI) are all examples of medical imaging. However, because of their broad availability and capacity to create high-resolution images of normal anatomic structures and diseases, CT and MR imaging are the most extensively employed procedures [KEBS15].

3.2.1 Brain Imagining Modality

Computed Tomography (CT)

It is a computerized x-ray imaging technology in which a narrow beam of x-rays is targeted at a patient and speedily rotated around the body, producing signals that the machine's computer processes to create cross-sectional images, or "slices." These slices are referred to as tomographic pictures, and they can provide a doctor with more information than traditional x-rays. After the machine's computer collects a number of successive slices, they can be digitally "stacked" together to create a three-dimensional (3D) image of the patient, allowing for simpler identification of basic components as well as any tumors or anomalies. An example for its device, and the output is shown in the figure below.

Figure 3.6: Two images showing both the device and result of CT device.

CT scans can detect potentially fatal illnesses like hemorrhage, blood clots, and malignancy. A timely diagnosis of these illnesses could save a person's life. CT scans, on the other hand, use x-rays, and all x-rays emit ionizing radiation. In living tissue, ionizing radiation can produce biological consequences. This is a risk that rises with the number of exposures accumulated throughout a person's lifetime.

Positron Emission Tomography (PET)

PET scans are imaging tests that can reveal the metabolic and biochemical function of your tissues and organs. A radioactive substance (tracer) is used in the PET scan to show both normal and abnormal metabolic activity. A PET scan can typically reveal aberrant tracer metabolism in diseases before other imaging procedures, such as computed tomography (CT) and magnetic resonance imaging (MRI), can detect the disease (MRI).

Noncancerous illnesses can mimic cancer, and certain malignancies do not show up on PET scans, so they must be read cautiously. PET scans can be used to diagnose brain cancers, Alzheimer's disease, and seizures, among other conditions.

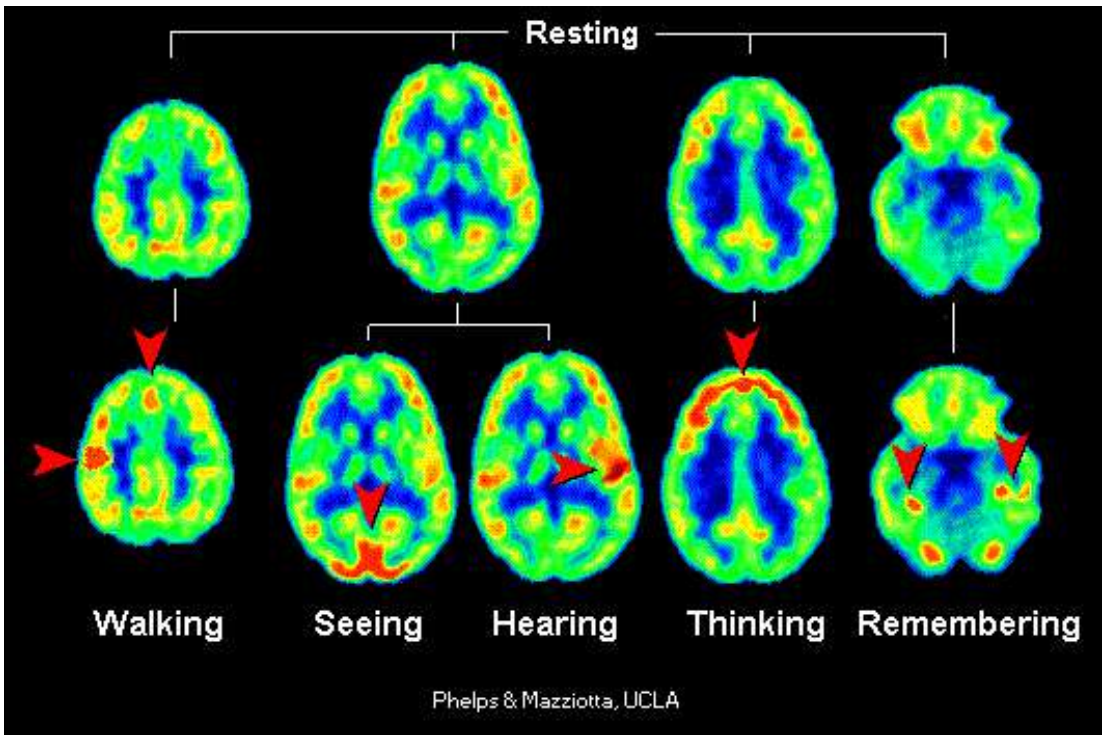


Figure 3.7: Brain PET Scan

PET scanning necessitates the use of a small amount of radioactive material (tracer). This tracer is injected into a vein (IV) on the inside of the elbow. Alternatively, you can inhale the radioactive substance as a gas.

The tracer passes through your bloodstream and settles in your organs and tissues. The tracer aids the clinician in gaining a better understanding of certain areas or disorders. The risk of negative effects from the radiation is low, but it is not well efficient for full detection for what needed.[Bra19]

Single-Photon Emission Computed Tomography (SPECT)

A nuclear imaging scan that combines computed tomography (CT) and a radioactive tracer is known as SPECT. Doctors can examine how blood travels to tissues and organs thanks to the tracer. A tracer is injected into your circulation before the SPECT scan. The tracer is radio labeled, which means it produces gamma rays that the CT scanner can detect. The information emitted by the gamma rays is collected by the computer and displayed on the CT cross-sections. These cross-sections can be reassembled to generate a three-dimensional representation of your brain.

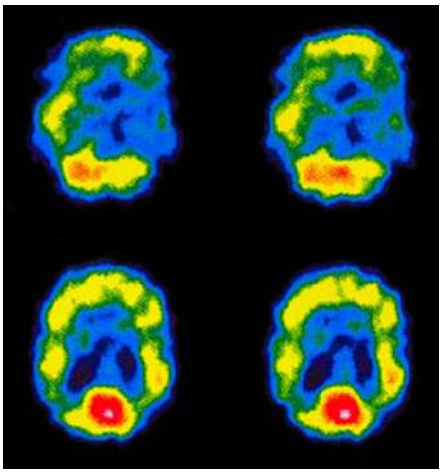


Figure 3.8: A patient with uncontrolled complex partial seizures undergoes a SPECT scan. The left side of the brain's temporal lobe has less blood flow than the right, indicating to the surgeon that a nonfunctioning portion of the brain is producing the seizures.

Magnetic Resonance Imaging (MRI)

It is a non-invasive imaging technique that creates three-dimensional anatomy images in great detail. It is frequently used to detect diseases, diagnose them, and track their progress. It is based on cutting-edge technology that excites and detects changes in the rotational axis of protons in the water that makes up biological tissues.

They differ from computed tomography (CT) in that they do not employ x-rays, which emit harmful ionizing radiation. MRI images the brain, spinal cord, and nerves, as well as muscles, ligaments, and tendons, considerably more clearly than normal x-rays and CT; as a result, MRI is frequently used to image knee and shoulder problems.

An example of MRI scan images of a patient obtained from [WNS+13] is visualized in Figure 3.9.

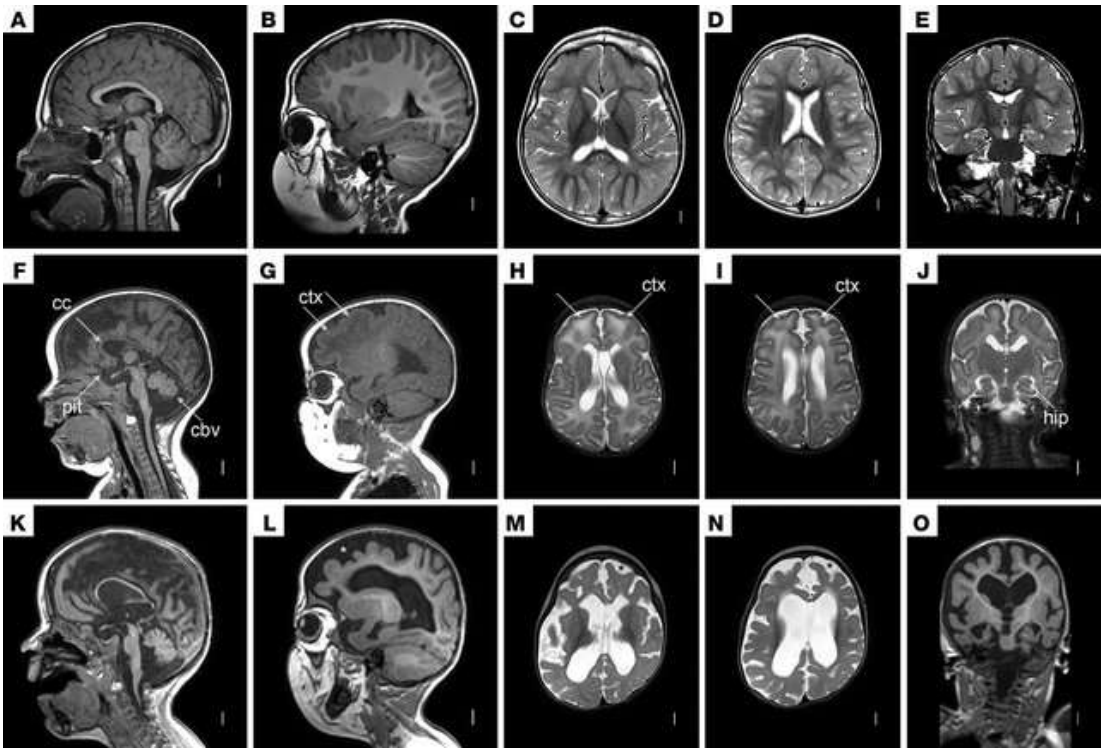


Figure 3.9: MRI scan Images

3.2.2 Approaches for Brain Tumor Segmentation

A generic approach for the full process of brain tumor detection is shown in the below figure. Due to the previous explanation about the modalities used for body

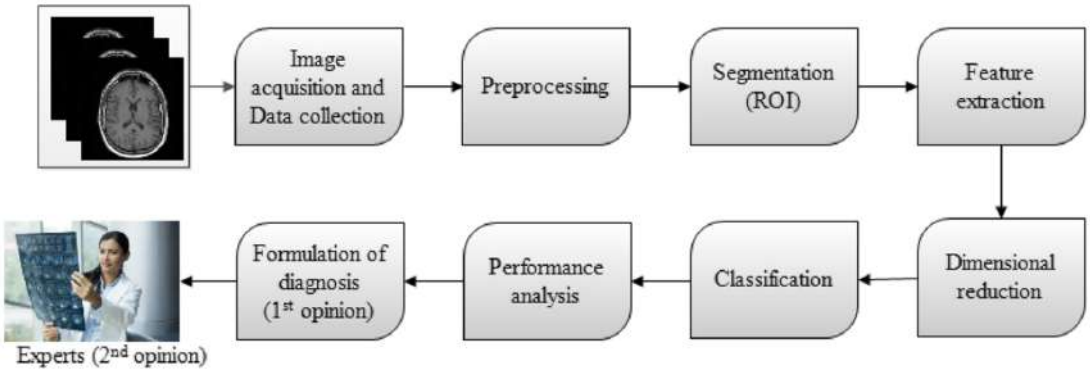


Figure 3.10: Process flow of a brain tumor detection system

scanning and specific, the brain. The MRI is the best used for its lower bad side effects, at the same time most of the studies apply their algorithms to this type of imaging.

Treating our MRI as an Image, to the segment it, we are dealing with approaches for image segmentation, except for the fact that these images have a more complicated structure. The technique of partitioning a digital image into several segments (sets of pixels, also known as superpixels) is known as image segmentation [Wik22].

The purpose of segmentation is to make an image more understandable and easier to evaluate by simplifying and/or changing its representation. Objects and boundaries (lines, curves, etc.) in images are often located via image segmentation. Image segmentation, in more technical terms, is the process of giving a label to each pixel in an image so that pixels with the same label have similar visual qualities.

Image segmentation, in general, has several methods, and mostly all of them were applied in the domain of brain tumor segmentation, you may find several studies related to segmenting tumor based on region segmentation including histogram threshold, watershed and many others like those appearing in [TS02, OK14, MJF12], other studies figure out the segmentation by the supervised machine learning and they are many [AE18, GGJY20, MDCA19, PPAS16]. Also brain tumor was detected by methods considered as unsupervised methods like Fuzzy-C means [S+15], K-means [PJP14], Singular Value Decomposition [MPF+18].

[GMS13, Won05] discuss the several approaches and techniques used for the segmentation and they classify it as threshold-based, boundary-based, region-based, clustering-based methods, Neural Networks, and many more. As an example, threshold-based approaches work on the assumption that pixels within a particular range belong to the same class [Ots79, SANI16]. While the boundary-based, it is based on the concept that pixel attributes drastically change at the intersection of two regions [GSO00, XP98]. Region-Based techniques presume that a region is made up of pixels with comparable attributes that are adjacent to one another [BAN17, SLW+00].

The authors in [TOC10] used level sets to try to determine a global threshold for segmenting tumor and non-tumor regions. To complete the operation, it only requires a zero level set to be fed into it. However, if the difference between the intensity levels of tumorous and non-tumorous regions is smaller, its efficacy is called into question. When the intensity of image pixels is low contrast and not homogeneous, or when there is a lot of noise, the global thresholding strategy performs poorly.

In [ASYF20] the authors provide an automated method for brain tumor segmentation and classification. Different kernels of the SVM classifier are used to categorize the different stages of malignant or non-cancerous pictures after partitioning the region of interest (ROI) comprising intensity, shape, and texture. Based on AUC (Area Under Curve) and ACC (accuracy) performance indicators, the proposed technique has been cross-validated on three different datasets: Local, Harvard, and Rider. The results show that the proposed method is effective.

Another approach by [TS19] uses Adaptive Differential Evolution with the Lévy Distribution strategy. During multi-level thresholding, DE is employed to keep the

balance between exploration and exploitation. Brain MRI images were segmented using the multi-level thresholding method.

The authors in [uAJC14], had proposed that after noise removal and feature extraction, they use the ensemble basis SVM classifier to classify the tumorous and normal brain images. After that, the tumor region is extracted using the fuzzy c-mean (FCM) clustering algorithm. Many patients' datasets were utilized to test this technique, which was acquired from the Abrar MRI & CT Scan center and the Holy Family hospital in Rawalpindi. The method claims to have a 99 percent accuracy rate.

This was a brief related work for the brain tumor segmentation by the use of several techniques applied to the image segmentation field in general, for more related approaches, the reader is guided to check the following references, [BSAD21, KS18, WBV19, ZSS+19].

Since our main focus in this field is on graph-based segmentation for a brain tumor, we summarized below the related work in recent years by the use of the graph concept.

One of the most graph-based approaches used for image segmentation is that proposed by [FH04], the method specifies a predicate for calculating the evidence for a boundary between two sections of an image. An efficient segmentation algorithm is created based on the provided criteria. It takes into account two variables to assess the evidence for the boundary. The paper defines two criteria for determining if evidence for a boundary between components or partitions exists. The internal difference (the component's largest weight in the MST) and the difference between the two components are the two criteria. The algorithm's ability to maintain information in "low variability regions while discarding details in high variability regions" is a unique feature. The program effectively separates the images and generates segments that capture the global attributes.

As recent work in [MHH+22] that depends on the minimum spanning tree, without tweaking settings, performs interactive segmentation based on the minimal spanning tree Pre-processing, graphing, generating a minimal spanning tree, and a newly designed method of interactively segmenting the region of interest are among the procedures. To reduce noise from 2D images, a Gaussian filter is utilized in the preprocessing step. The pixel neighbor graph is then weighted by intensity differences, and a minimum spanning tree is created as a result. The image is imported into an interactive window where the tumor can be segmented. By clicking to split the minimal spanning tree into two trees, the region of interest and the background can be selected. The region of interest is represented by one tree, while the background is represented by the other. Finally, the two trees' segmentation is represented graphically. The drawback of this method is its requirement for user interaction, while we are trying to obtain results simpler. Their approach is summed up in the figure below.

Graph Coloring Approach was used in [BMM21] where each pixel in the brain

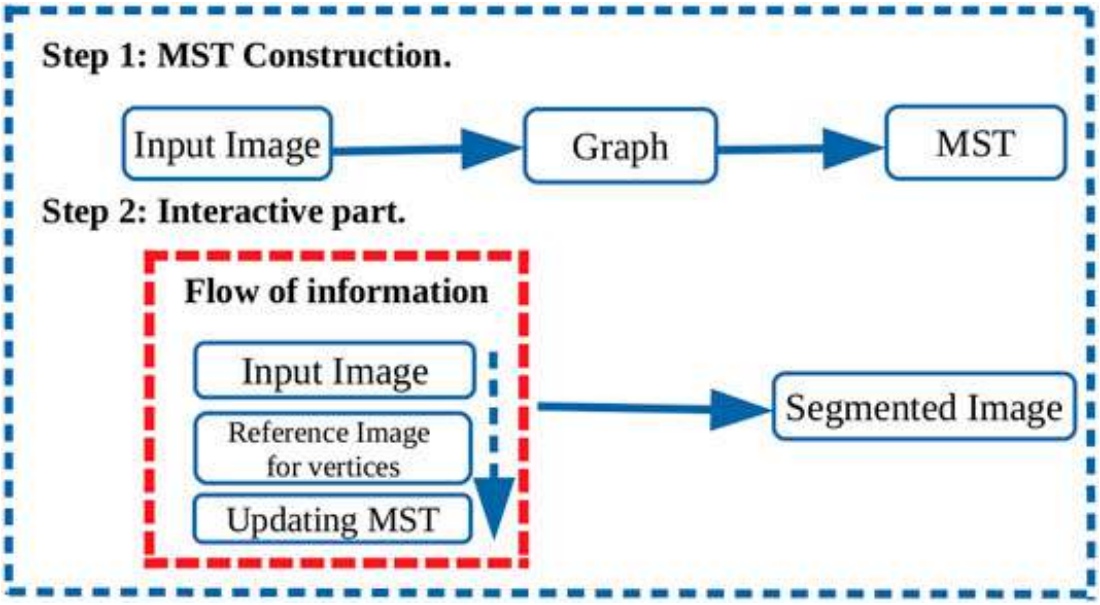


Figure 3.11: Schematic flow diagram shows the steps involved in the segmentation process.[MHH+22]

image is treated as a node in the graph, and the difference in brightness between a couple of pixels is treated as an edge. This approach was used on T1enhanced magnetic resonance images of patients with low and high grades. Because a stiff graph was required for graph coloring, edges must be classified as present or nonexistent using a threshold. The value of this threshold affects image segmentation accuracy, so selecting the best threshold was critical. The drawback in their work was the dependability of accuracy on the threshold of edges.

Another widely spread approach related to the Graph-Cut (GC) field for image segmentation. A graph converts pixels or areas of the source image into graph nodes. The segmentation problem can then be turned into a labeling problem, which requires assigning the appropriate label to each node based on its properties. The Markov random field (MRF) has been effectively employed in computer vision and machine learning to model pixel contexture information. This contextual information serves as a means of getting image attributes. The MAP–MRF framework [GG86] formulates the labeling problem as a graph-based minimization problem when combined with Bayesian maximum a posterior (MAP) estimation. GCs offer a versatile optimization method for solving the minimization problem with high computing efficiency.

A simple two-dimensional (2-D) image example is shown in Figure 3.12, in object segmentation, there are two specific terminal nodes named source s and sink t that represents "object" and "background" independently.

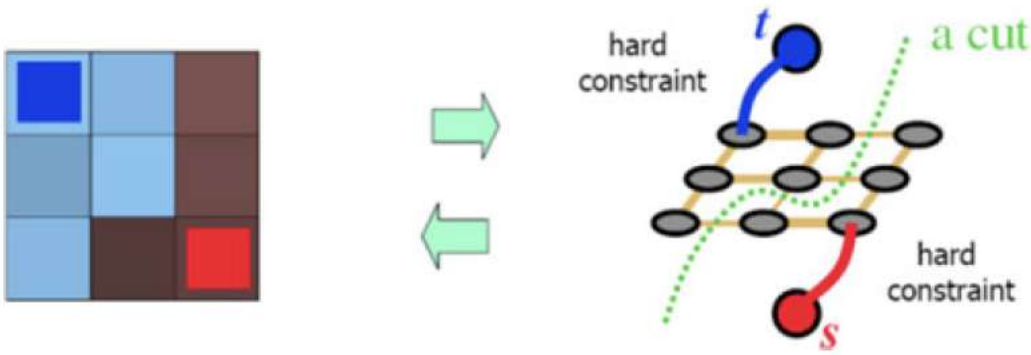


Figure 3.12: Diagram of graph building for a simple 2-D image. (This figure is based on Boykov's ECCV 2006 tutorial. Tutorial)

An s/t cut in a graph G is a division of V into two disjoint subsets S and T, with all object voxels connected to an object terminal node s and all background voxels connected to a background terminal node t. The goal is to select the best cut that will produce the greatest results based on the segmentation requirements. A cut with the lowest cost is both the cheapest and best cut.

Detailed related work for the graph-cut medical segmentation could be found here: [BP12, CG12, CP18].

Chapter 4

New Descriptors' Combination for 3D Mesh Correspondence and Retrieval

" The price of success is hard work, dedication to the job at hand, and the determination that whether we win or lose, we have applied the best of ourselves to the task at hand."

– Vince Lombardi

4.1 Introduction 38
4.2 Proposed Approach 39
4.2.1 Interest Point Detection 39
4.2.2 Feature Descriptor Formulation 40
4.2.3 Feature Matching 42
4.3 Experimental Results 44
4.3.1 Interest Point Detection Experiments 44
4.3.2 Matching Experiments 46
4.4 Conclusion 49

Summary

This chapter introduces our new novel method that finds out a formulates a new combination of descriptors, leading to a well optimal matching and retrieval. It conducts matching algorithms, studying the difference between them in obtaining the best correspondence.

4.1 Introduction

With the large spread of 3D models over the Internet and databases, classification, and identification of them have become an essential task to do. At the same time, 3D models are huge in their structure because of the speedy technology acquisition and storage, which encourages developing methods to have the ability to access it simpler and faster.

Shape descriptors are used to lay out the distinctive description, meaningful, and compact content of 3D models. The properties of invariance to a class of transformations, discrimination, stability, and completeness should be verified by these descriptors.

An instance of shape descriptors is the feature-based approach, which that becomes popular in fundamental problems of computer vision and shape analysis, as well as the correspondence between these descriptors of several models considered as the key ingredient in a range of applications including shape retrieval [BS12, LGB+13, TV08], classification [Tab11], recognition [Abd13, BPK16, SAO17], and became an essential way for the search engines.

Computing local shape features [HPPLG11] for every model and assessing a similarity measure for the pairs, optimizing it with a cost function to have the optimal correspondence, this is what feature correspondence tries to output. Several studies [DCG12, GBS+16, MP07, RBRY19, SSC20, TG12, VKZHCO11] evaluate different types of detectors and descriptors, testing them on different databases, and proving the effectiveness of one on another.

The matching problem could be seen as a graph matching problem since the graph is considered a powerful data structure in such problems, this is seen clearly in [TKR08, ZTZ15].

Our work in this chapter is twofold: first, we try to formulate a new combination of the local descriptor, that consists of a simple, yet powerful set of descriptors; second, we evaluate this descriptor by a comparative study on shape correspondence through examining two different matching algorithms, as well on shape retrieval system checking the correctness of retrieved meshes. And while most of the used similarity measures in the state-of-art are the Euclidean distance, we tend to use the cosine similarity measure for comparing the set of descriptors between every two models, which shows its powerful results in [BZA+18] and ours. Our tests are done depend-

ing on a comparative study as well for the parameters to be used in the 3D-Harris detector [SB11].

Our main motivation is to represent 3D models in a significant way, that is fast, reliable, distinctive, as well as optimal in sense of recognizing models, whether for matching, classifying, or segmenting problems.

The proposed approach that we formulate takes the advantage of extracting well-distributed key points, that minimize the number of vertices to deal with, followed by combining a discriminating descriptor with optimal storage allocation and size.

The organization of this chapter is as follows. Section 4.2 shows the formulation of our problem. Section 4.3 shows the experimental results, ending up with Section 4.4 that concludes the work, followed by the main contributions.

4.2 Proposed Approach

We start our method by modeling the 3D mesh as an undirected graph G = (V, E), where V is the set of vertices, and E is the set of edges E $\subset$ V x V . This is followed by three key components: interest point detection, descriptor extraction, feature matching.

4.2.1 Interest Point Detection

The intended detector to be used in our work is the 3D Harris detector. Harris 3D works in a vertex-wise manner to compute the Harris response for each vertex [Sip, SB11]. The author follows the criterion below: for a vertex $v_i \in$ V, neighborhood N_{v_i} is defined in two different ways since we perform a comparison to check the best way.

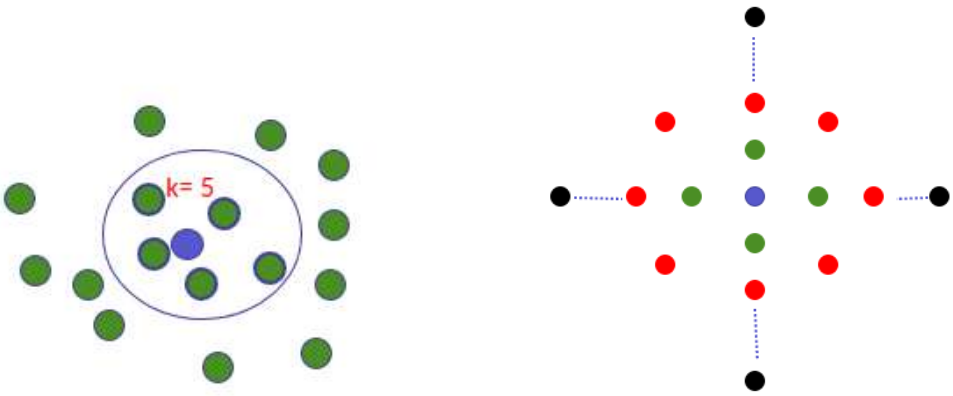
The first consideration for the neighbors is by selecting a fixed number of points that are closest to the point v_i depending on the computed Euclidean distance between the desired vertex and all others, brute-force k-nearest neighbors algorithm, and an illustration is shown in Figure 4.1 (a). Another way to compute the neighborhood of a vertex is by the adaptive k-ring Delaunay where the distance from a point v to $ring_K(v)$ by:

$$d_{ring}(v, ring_K(v)) = max_{w \in ring_K(v)} ||v \neg w||_2 \tag{4.1}$$

and the neighbourhood size of a point v is given by:

$$radius_v = \{K \in \mathbb{N} \mid d_{ring}(\, v, ring_K(\, v)\,) \geq \delta \quad \text{and} \\ d_{ring}(\, v, ring_{K \neg 1}(\, v)\,) < \delta\}$$

where δ is a fraction of the diagonal of the object bounding rectangle, its illustration is shown in Figure 4.1 (b).



(a) K-nearest neighbours

(b) K-closest rings around v, our desired vertex in blue

Figure 4.1: Neighborhood definition

A centroid for N_{v_i} is computed to translate all vertices V, where the centroid coincides with the origin of the coordinate system. After this, computing the fitting plane is done using Principal Component Analysis (PCA) on the set of points and choosing the eigenvector with the lowest associated eigenvalue as the normal of the fitting plane. This is followed by rotating the set of points until the normal of the plane coincides with the z-axis. Finally, the resulting xy plane (2D projection) is used for calculating the derivatives. These derivatives are evaluated using a six-term quadratic surface (parabolic) fitted to the set of transformed points. We refer the reader to the detailed work and formulas [SB11].

We mentioned the methods used in computing the neighborhood due to their importance in our study, as well as the importance of the strategy followed for selecting interest points, where two approaches defined by the authors were implemented, making the comparison part more fruitful. Each vertex has a Harris operator value, so the interest points are those with the highest response to a constant fraction, or those obtained by the clustering algorithm that gives well-distributed interest points.

4.2.2 Feature Descriptor Formulation

In this part, we come up with a local feature descriptor that is a combination of three different measures. After computing each of these measures on the interest points extracted, we formulate our new descriptor by compacting these values, obtaining a *compact feature vector* of three parameters for each keypoint. In what follows, we present geometrical details about the shape descriptors in use.

Gaussian Curvature

GC: It is obtained by multiplying the principal curvatures at a given point.

$$\mathcal{GC} = \kappa_1 \kappa_2 \tag{4.2}$$

A sample visualization for it is shown in the figure below.

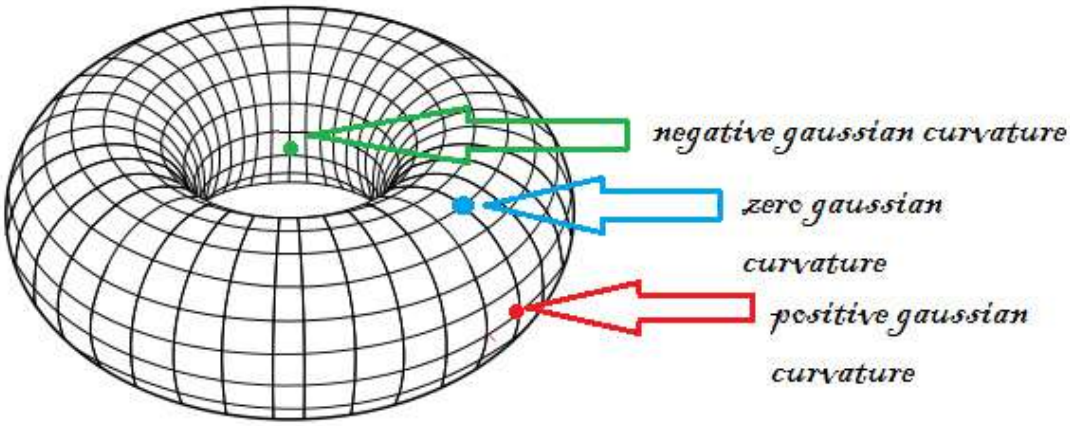


Figure 4.2: Gaussian Curvature

Curvature Index

CI: It depends on the principal curvature factors and can be calculated by:

$$\mathcal{CI} = \sqrt{\frac{\kappa_1^2 + \kappa_2^2}{2}} \tag{4.3}$$

Shape Index

SI: It is a value that represents the topology of the local surface using the principal curvatures. It is calculated by the following formula:

$$\mathcal{SI} = \frac{2}{\pi} \tan^{-1}(\frac{\kappa_2 + \kappa_1}{\kappa_2 - \kappa_1}) \tag{4.4}$$

Where κ_1 and κ_2 are the principal curvatures with $\kappa_2 \geq \kappa_1$.

The study [TG12] shows the good results obtained for both Gaussian Curvature and Curvature Index, same for Shape Index it had been proved in [GBS+16] to be a powerful descriptor. We believe that combining these features will assist in avoiding the little gap in each.

After computing these features, we end up with a new descriptor that consists

of a matrix of dimensions *kp* * 3 for *M*, the key points extracted from the mesh we are dealing with.

4.2.3 Feature Matching

Matching two models through the detected and described features, is set to be the most important part of computer vision applications, such as classification, retrieval, recognition, and others. Our task in this paper is to establish a one-to-one correspondence between two sets of descriptors.

Finding such mapping is a challenging one, due to determining the similarity measure to be used, as well as the algorithm for this matching.

Similarity Measure

The similarity measure between two feature vectors is described as the distance between these vectors, the nearer distance we have, the more similar shapes they are.

The metric we tend to use in this paper is the cosine distance, which shows effective results in this domain over the Euclidean distance and it is shown clearly in [MA18, PAH+17, RAA+19, TLFF17] if the two similar models are far apart by the Euclidean distance (due to the size of the model), the chances are they may still be oriented closer together.

Cosine distance is obtained by the following equation

$$CosineDistance = 1 - CosineSimilarity \tag{4.5}$$

Where the cosine similarity is calculated by the dot product over the magnitude of the two vectors,

$$\cos\theta = \frac{\sum_{i=1}^{n} des_{i1} des_{i2}}{\sqrt{\sum_{i=1}^{n} des_{i1}^2}\sqrt{\sum_{i=1}^{n} des_{i2}^2}} \tag{4.6}$$

(*n, des*) represent the number of key points and their descriptors, respectively.

In figure 4.3 we show a simple visualization with a general formula for the cosine similarity metric, where the cosine distance is calculated between two shapes by the given equation.

Obtaining this distance matrix is done through the bipartite graph calculation shown in Figure 4.4.

We compute the cosine distance between each descriptor in model M_1 with all descriptors from the second model M_2 ending up with a square distance matrix D_M with kp_1 * kp_2 dimension. Distance matrix shown in Equation 4.7 is the input for our

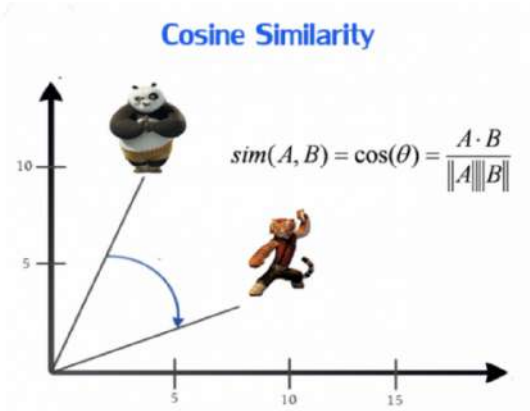


Figure 4.3: Cosine Similarity between two models.

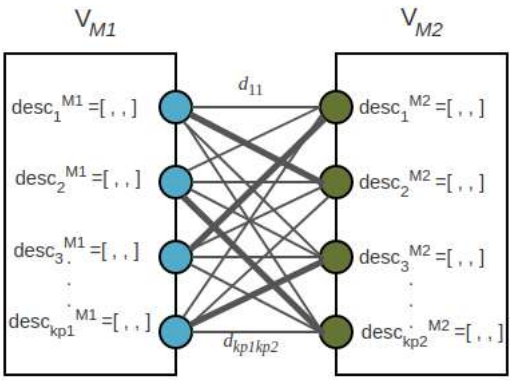


Figure 4.4: Bipartite graph for finding the similarity measure between descriptor vectors. V_{M1} and V_{M2} are the two sets of interest points with their corresponding $featurevector$. d_{ij} is the weighted link between every two vertices with cosine similarity as a value.

next step.

$$D_M = \begin{bmatrix} d_{11} & d_{12} & d_{13} & .... & d_{1j} \\ d_{21} & d_{22} & d_{23} & .... & d_{2j} \\ . & . & . & .... & . \\ . & . & . & .... & . \\ d_{i1} & d_{i2} & d_{i3} & .... & d_{ij} \end{bmatrix} \tag{4.7}$$

Correspondence Matching Algorithm

After having the distance matrix D_M that shows the cosine distance between every two key points, it is time to get the one-to-one correspondence between these points. We do this by checking the effectiveness of two algorithms, brute-force matching usually combined with the ratio test and the Hungarian one.

Brute-force Algorithm In this approach and using the built-in library from *skimage*, for each descriptor in the first set, this matcher finds the closest descriptor in the second set. The ratio between the distance of the closest and second-closest corresponding feature determines whether the potential pair is accepted or not. As most used, especially in SIFT descriptor, the threshold is set to 0.8, where when getting a ratio greater than the threshold, the match is omitted. And as mentioned before, the desired similarity measure is set to cosine distance.

Hungarian Algorithm The Hungarian algorithm [Kuh55] computes a complete matching of the bipartite graph such that the distance matrix obtained from (7) is our input, where the distances are the entries, to find the optimal correspondence between the key points of the two models. If the cosine similarity between two key points is 1, this means that they are similar models.

We also propose a constraint for the good and bad matches, after obtaining the corresponding mapping, if the cosine similarity of those retrieved mapped keypoints is greater than or equal to 0.8, this is considered a good match, otherwise, it is bad.

4.3 Experimental Results

We did our experiments on 100 models obtained from two datasets published online, [Pap14, SP04], and we arrange them into nine classes, each containing different poses of a similar model. The models are cat, chair, dolphin, gun, hands, horse, lion, person, and spider poses.

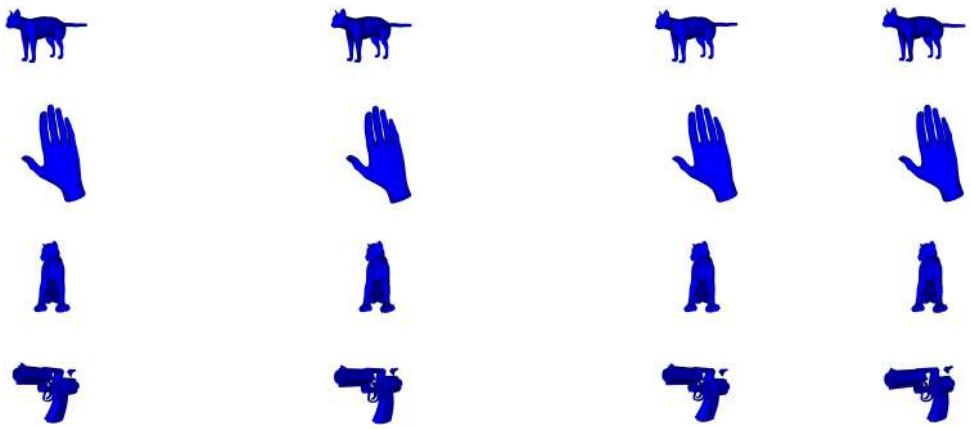
The proposed approach was implemented in Python and tested on Intel Core i7 Ubuntu 20.04.1 LTS with 8 GB memory.

4.3.1 Interest Point Detection Experiments

The purpose behind these experiments is to check the effectiveness beyond the method chosen for extracting the key points. As mentioned in sub-section 3.1 we tried to extract the key points using the KNN (k = 5), and K-ring Delaunay (δ = 0.025) for neighborhood computing with 0.1, 0.2 fraction ratio and 0.01, 0.02 for cluster threshold.*vedo* [MJD+22] software was used for visualization.

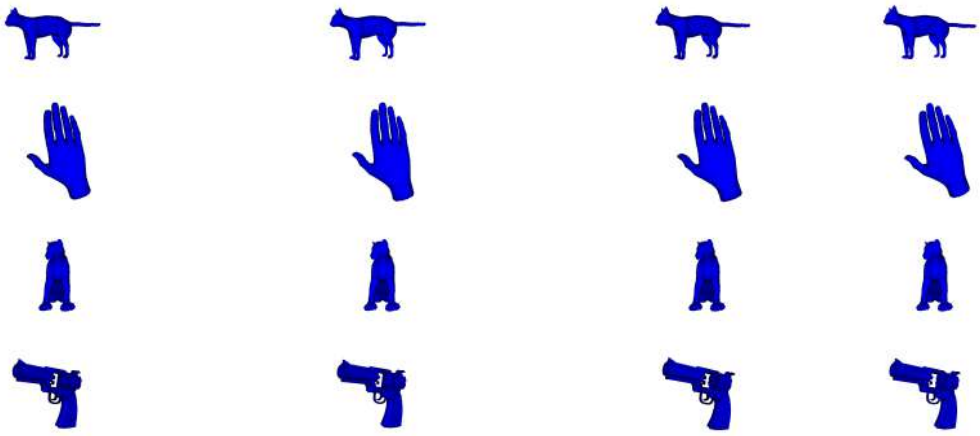
Figures 5,6 show the representation of interest of points on the 3D models, and it's obvious how these points are homogeneously distributed in the clustering case, contrary to that in the fraction, as well as it's well-noted visually that in fraction despite the way of computing neighborhood the same number is given, while it is not that case in clustering. Table 4.1 shows an instance for two models, two different poses, and their numerical results for keypoints extraction.

These results are a sample from the 100 models working on, it shows that the same number of key points is extracted while working in the fraction case, despite



(a) 10% KNN (b) 20% KNN (c) 10% Ring (d) 20% Ring

Figure 4.5: Examples of four 3D models with the key points detected in black on each using the Fraction-method.



(a) 0.01 KNN (b) 0.02 KNN (c) 0.01 Ring (d) 0.02 Ring

Figure 4.6: Examples of four 3D models with the key points detected in black on each using the Cluster method.

the neighborhood method computation, while in the cluster case there is a difference, even though it might be a slight difference.

Since the choice of keypoints method plays a vital role in the rest of the procedure, we tend to continue working on the fraction method, which ensures robustness for the shape posing, and the 3D Harris detector has already been proved by their authors to be invariant under affine transformations.

Table 4.1: Numerical results for keypoints extraction, two poses of two models are shown

Models	KNN				K-Ring			
	Fraction		Cluster		Fraction		Cluster	
	0.1	0.2	0.01	0.02	0.1	0.2	0.01	0.02
Cat pose 1	**720**	**1441**	**1461**	**462**	**720**	**1441**	**1461**	**462**
Cat pose 2	**720**	**1441**	**1502**	**476**	**720**	**1441**	**1496**	**478**
Hand pose 1	**248**	**497**	**2348**	**1644**	**248**	**497**	**2337**	**1637**
Hand pose 2	**248**	**497**	**2349**	**1646**	**248**	**497**	**2349**	**1646**

4.3.2 Matching Experiments

After detecting the key points, the feature vector is extracted, to study how much our new descriptor combination is well enough we use first to check it with one-to-one correspondence and try both brute-force and Hungarian algorithms for getting this correspondence by checking how much these mappings are good. The other way is by retrieving models from the database, upon a query mesh, checking how close these retrieved are.

One-to-One Correspondence Experiments

Figure 4.7 shows an instance for two models from two different classes using the brute force matching algorithm, while Figure 4.8, for the same model, the Hungarian algorithm is tested and the results were revealed showing the correct matching in green lines while bad matches are in blue.

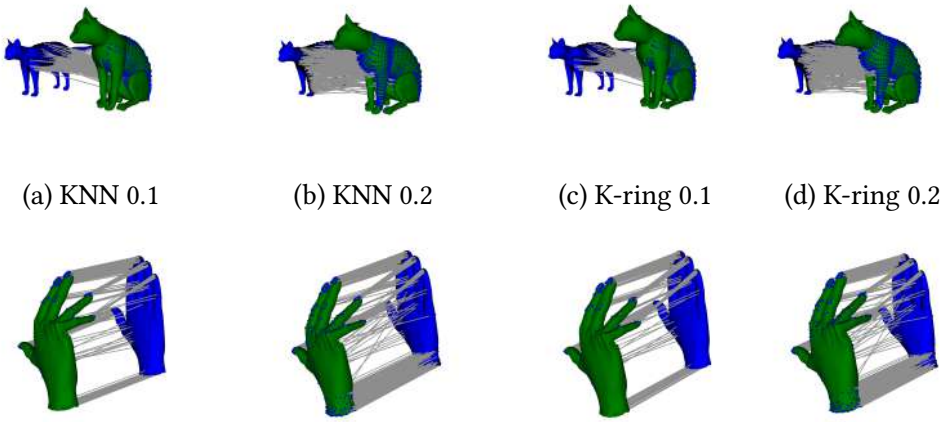
Other than the visualization part, the graphs in Figure 4.9 shows for each class in the database, the two models were matched one-to-one and their correct match with the number of key points is shown. Our results succeeded well in the Hungarian algorithm, where the correct matches are nearly full, the same model with different poses. In contrary to the brute force algorithm which lacks this.

Retrieval System

To investigate the strong discrimination in our proposed approach between classes of 3D models, we tend to compute a similarity matrix, **Cosine Distance Matrix** between all pairs in the database. The results are exposed in Figure 4.10.

Our classes are not fully equal, where some classes have 10 models, the others are with 12, so definitely the matrix will not be symmetric. The more the distance tends to zero, the more the shapes are correctly matched.

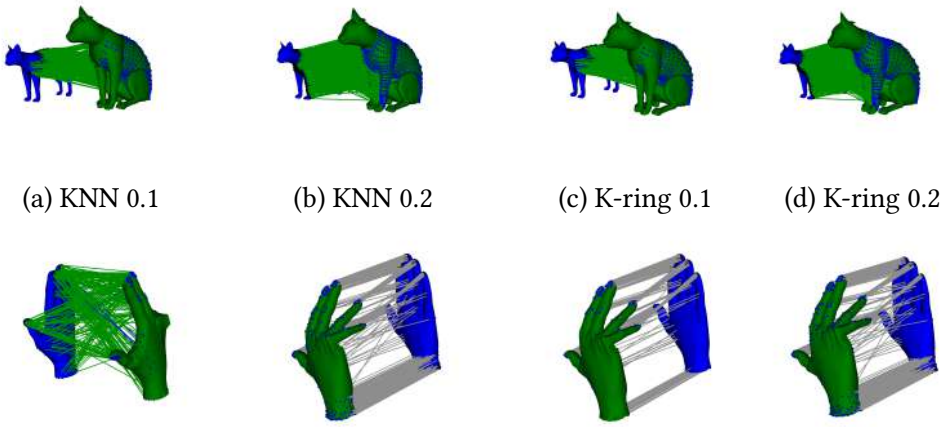
We believe that our approach has retrieved a strong similarity, indeed there is a high similarity between different models from different classes, and this refers to the concept of sharing the same geometric aspect despite the semantic one, like the cat



(a) KNN 0.1 (b) KNN 0.2 (c) K-ring 0.1 (d) K-ring 0.2

(e) KNN 0.1 (f) KNN 0.2 (g) K-ring 0.1 (h) K-ring 0.2

Figure 4.7: Two different poses were matched using the Brute-force algorithm for cats in the first row, and hands in the second row, with the different parameters for neighborhood calculation



(a) KNN 0.1 (b) KNN 0.2 (c) K-ring 0.1 (d) K-ring 0.2

(e) KNN 0.1 (f) KNN 0.2 (g) K-ring 0.1 (h) K-ring 0.2

Figure 4.8: Two different poses were matched using the Hungarian algorithm for cats in the first row, and hands in the second row, with the different parameters for neighborhood calculation, blue and green lines refer to bad and good matches respectively.

model and the lion one. This somehow represents a drawback in our system, which we will be optimizing in the future work,

Figure 4.11 shows several queries done to retrieve similar 3D objects from the database. The first model is the query and the first retrieval model with cosine sim-

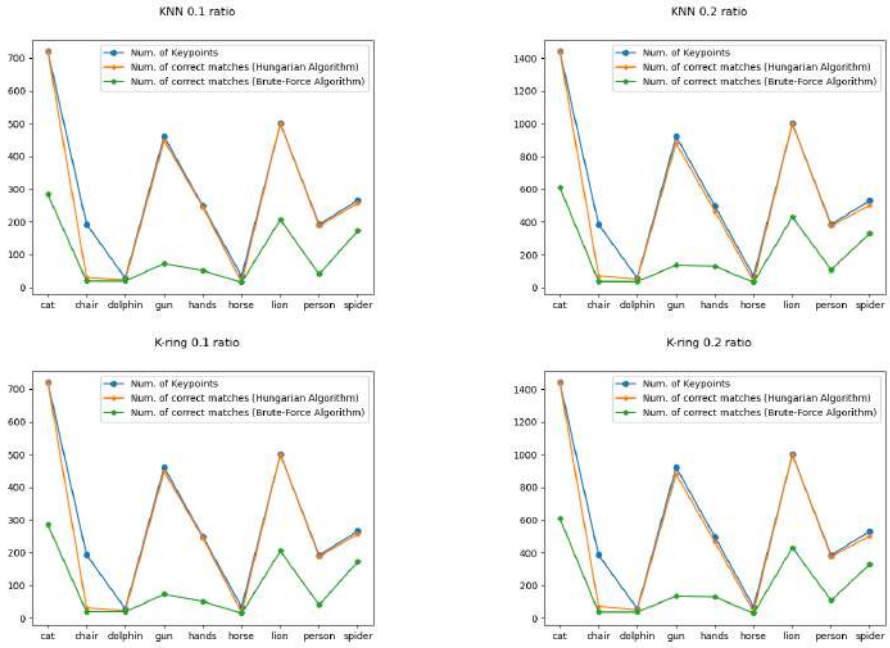


Figure 4.9: One-to-One Correspondence match between two meshes from each class, showing a number of key points, as well the correct matches retrieved by Brute-force, and Hungarian algorithms.

Figure 4.10: Cosine Distance Matrix for the used 100 Model Database classified into 9 classes

ilarity equals 1 and the others are top retrieved with the constraint that the cosine similarity should be greater than or equal to 0.8, saying that this is a similar object.

The default retrieving is for the top 10 despite having more for the chair pose we can see that it retrieves them all similarly for the hand and dolphin models in (c) and (d) respectively, while for the gun model our approach retrieved 8 out of 12 models, the same as for the person model. We believe that the way reached to describe the key points with the similarity measure we took, has shown a very well-noted result.

4.4 Conclusion

Following the several studies mentioned in the introduction that show the powerful detector 3D Harris did in [SB11], we tend to conduct our own comparison of the clustering and fraction methods to determine which method best suits our application.

At the same time, the validity of mixing descriptors in both feature vector and histogram-based in the work [TG12], which, in addition to demonstrating that feature vector is superior to histogram-based, went through the process of creating our own new descriptor feature vector. And since in our point of view that cosine distance is more efficient than euclidean distance we used it to be our similarity approach between two models.

As conclusion, this chapter presents a new combination of descriptors that shows a good performance, the obtained results are very satisfactory.

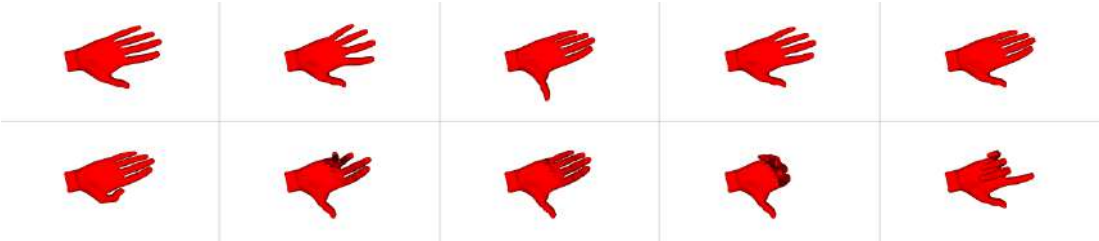
Our proposed approach mixes a wide used detector with the new combination of our descriptor, as well ending up with the use of the Hungarian algorithm for matching bipartite graphs that shows comparable efficiency between its strong polynomial time-bound and the combinatorial complexity of a brute-force algorithm.

These experiments are desired to be tested on larger databases in order to check whether it efficiently gives the same results in optimal time. As well our future work is desired to introduce a multi-scale descriptor which we believe will enhance the good matches obtained, in addition to using more efficient optimization modeling due to the new studies obtained.

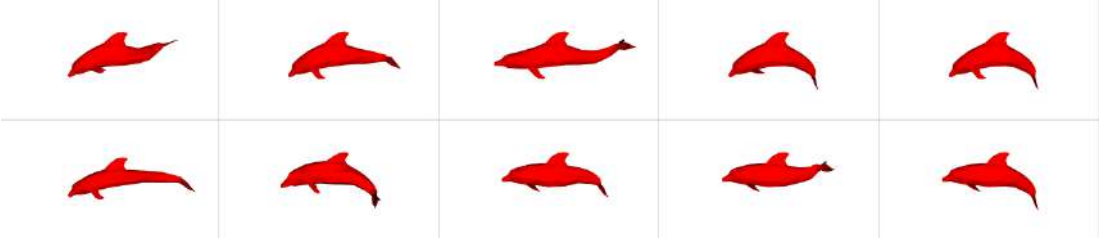
We believe in upcoming studies due to the enhanced optimizations to be done, we will end up with a more powerful descriptor that overcomes the main issues of the state-of-art descriptors.

(a) Chair Model

(b) Gun Model



(c) Hand Model



(d) Dolphin Model

(e) Person Model

Figure 4.11: Retrieval results of different queries

Chapter 5

3D mesh matching using surface descriptor and integer linear programming

"If you don't go after what you want, you'll never have it. If you don't ask, the answer is always no. If you don't step forward, you're always in the same place."

– Nora Roberts

5.1 Introduction 52
5.2 Proposed Framework 52
5.2.1 Descriptor map estimation 55
5.2.2 Feature Regions Detection 57
5.2.3 3D Surface Matching 59
5.3 Experimental data analysis and results 60
5.4 Conclusion 65

Summary

This chapter is composed of mainly published work in a conference, related to the use of Integer Linear Programming for surface matching.

5.1 Introduction

Matching and similarity between 3D shapes is a fundamental task for shape segmentation, classification, and retrieval, and all of these together with many others are important procedures in different areas such as computer vision, biomedical modeling, mechanical engineering, etc.

At the same time, the widespread of 3D data brings the need to develop algorithms in order to match and classify as well as recognize it. Unlike images and range scans, 3D models do not depend on the configuration of cameras, light sources, or surrounding objects (e.g., mirrors).

As a result, they do not contain reflections, shadows, occlusions, projections, or partial objects, which greatly simplifies finding matches between objects of the same type.

Descriptors whether local or global ones should capture the most significant features of the shape in order to represent it in a good manner.

Many approaches were proposed to capture the details of shapes, some lack the capacity in terms of huge memory required, others may be variant towards affine transformations, etc. And since this step is crucial for the dependence of other steps on it, in our work we proposed an efficient way to describe the 3D model, by modeling its surface as a weighted graph using the Gaussian function, followed by combining precise curvatures formulating compact feature vector.

In the following, we use the Louvain clustering method implemented by [BdLLC20] to segment our mesh into regions. These regions are set to be our subgraphs, and they are matched through the linear sum assignment problem [VGO+20].

The rest of this work is organized as follows. Section 5.2, presents the complete framework starting with surface modeling, followed by region segmentation, and ending this section with the 3D matching problem. In Section 5.3, we show our experimental results and the robustness of our proposition. The work is concluded in Section 5.4 with the main contributions.

5.2 Proposed Framework

In our framework, we tend to find matching between 3D models. We start by modeling the surface of the mesh as a weighted graph, where our main target is

to match sub-graphs obtained from the feature regions extracted depending on the descriptor map. Our descriptor map is defined by extracting several curvatures and formulating a compact feature vector.

The following Flow-Chart shows the overall of our approach, pursued by detailed clarifications.

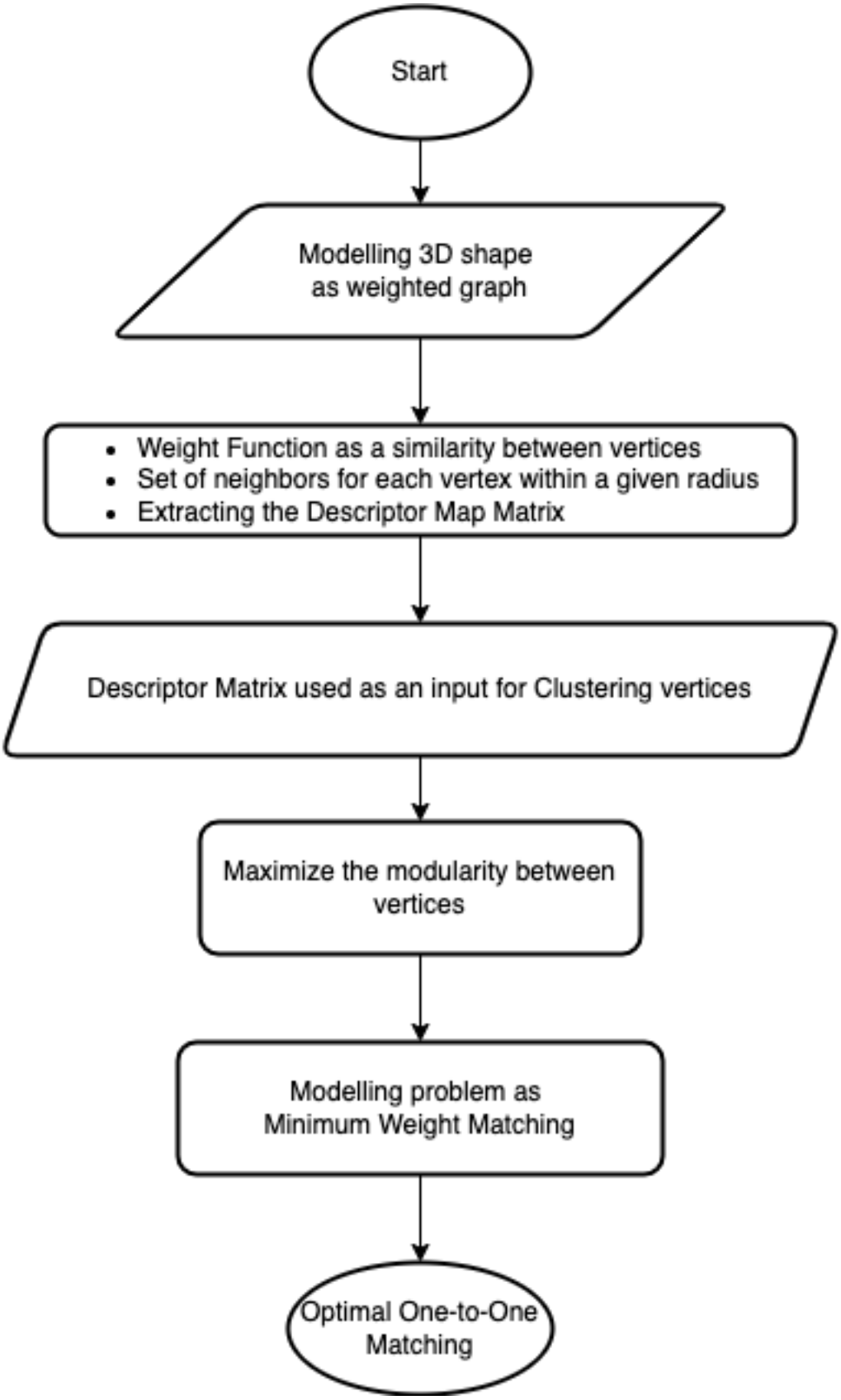


Figure 5.1: Flow-Chart of proposed approach

5.2.1 Descriptor map estimation

We start by representing the 3D surface as a weighted graph $\mathcal{G} = (V, E, \omega)$ where V is a finite set of vertices, E is a finite set of edges $E \subset V \times V$ and ω represents the weight function; it is a measure of similarity between vertices.

At the same time, we tend to obtain the set of neighbors $\mathcal{N}(v_i)$, for a given vertex v_i, where we find the neighbors within a given radius, and also the vertices lying on the boundary are included.

Figure 5.2 shows the construction of the neighborhood of a single vertex.

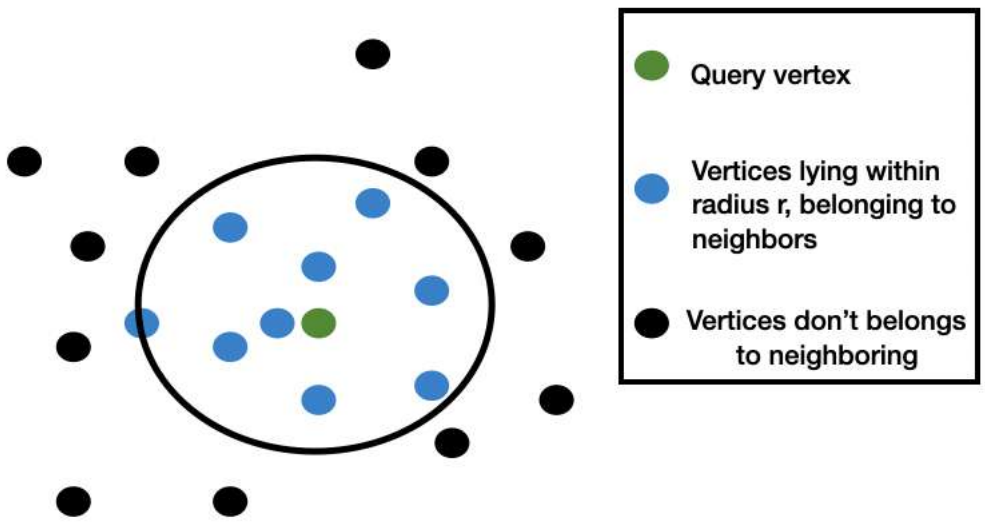


Figure 5.2: Neighboring Construction

The weight function between two connected vertices is calculated as follows.

$$\omega\left(\nu_i, \nu_j\right) = e^{-\frac{(d\nu_i - d\nu_j)^2}{2\Theta^2}} \tag{5.1}$$

Where d of any vertex represents its descriptor vector, it is important to mention that the calculated weight function is affected by the descriptors we are obtaining not by the usual Euclidean distance between their coordinates, while Θ is the standard deviation.

Each vertex has a compact feature vector, that called as a descriptor, and com-

puted by obtaining four curvatures as below:

$$\mathcal{SI} = \frac{2}{\pi}\tan^{-1}(\frac{\kappa_2 + \kappa_1}{\kappa_2 - \kappa_1}) \tag{5.2}$$

$$\mathcal{CI} = \sqrt{\frac{\kappa_1^2 + \kappa_2^2}{2}} \tag{5.3}$$

$$\mathcal{GC} = \kappa_1\kappa_2 \tag{5.4}$$

$$\mathcal{MC} = \frac{1}{2}(\kappa_1 + \kappa_2) \tag{5.5}$$

When computing a characteristic vector of four for each vertex, we will end up with a matrix $\mathcal{D}$ [n *4] where n is the number of vertices we have in our model, and four are the components of the descriptor.

We show in Figure 5.3 the representation of our descriptor compared to ground truth, and the proposed approach from [ECESAB20], which shows how the feature vector extracted by our approach is doing well in discriminating regions such as ground truth and is more obvious than the approach mentioned.

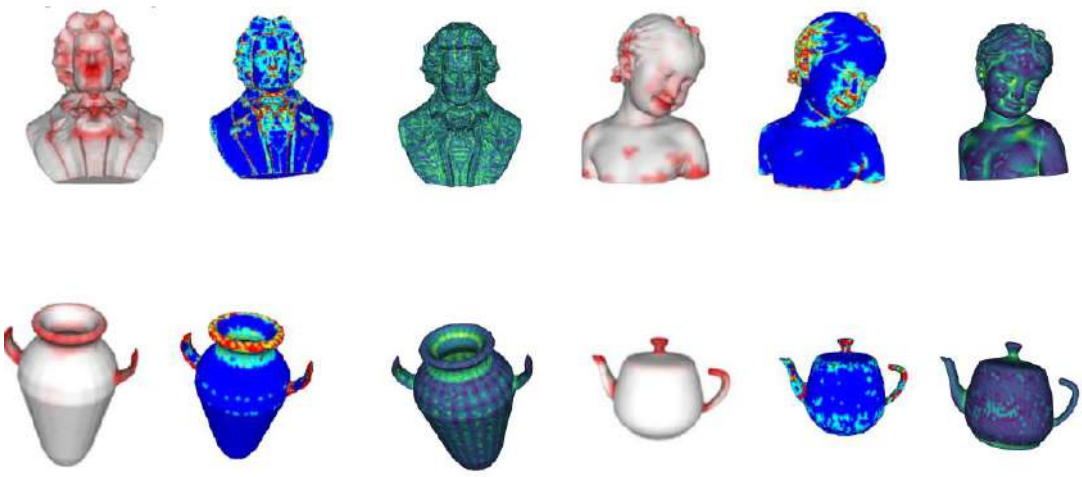


Figure 5.3: Descriptor Maps. Left to Right: Ground Truth, proposed approach by [ECESAB20], ours

In the images 5.4 as well to5.5, we show the descriptor map for different gestures of faces in several cases: fury, anger, smile, sad, cry, laugh, normal, and surprise, respectively. In addition to different poses for the elephant.

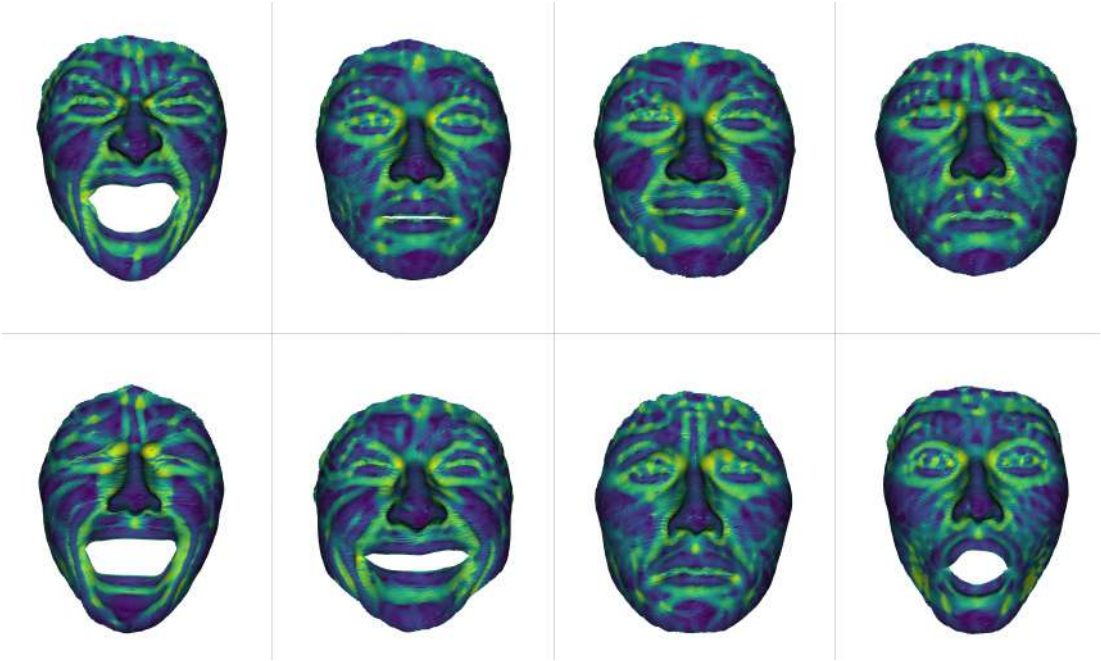


Figure 5.4: Descriptor Maps: Faces Reactions

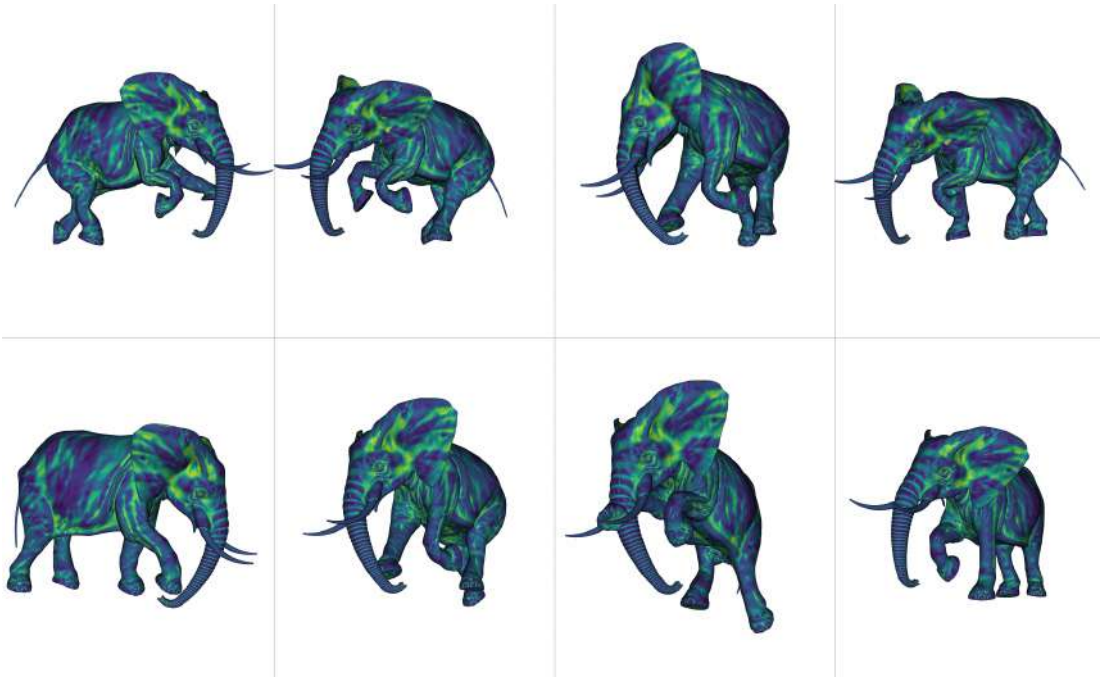


Figure 5.5: Descriptor Maps: Elephant Poses

5.2.2 Feature Regions Detection

Due to the large number of points the 3D model has, clustering these vertices is considered one of the important steps to be applied before matching these objects for

several reasons, the main one being to reduce the complexity of obtaining the desired results; this will give us a subgraph representation to deal with.

To do so, we cluster our vertices using the weighted graph obtained by computing the weights from the descriptors obtained.

Integer linear programming was used to model our problem, where we try to maximize the modularity between vertices. At first, each vertex has its own label, vertices similar to each other are merged, and labels are decreased.

The modeling is as follows:

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j), \tag{5.6}$$

where

- A_{ij} represents the edge weight between nodes i and j;
- k_i and k_j are the sum of the weights of the edges attached to nodes i and j, respectively;
- m is the sum of all of the edge weights in the graph;
- c_i and c_j are the communities of the nodes; and
- δ is Kronecker delta function ($\delta(x, y) = 1$ if $x = y$, 0 otherwise).

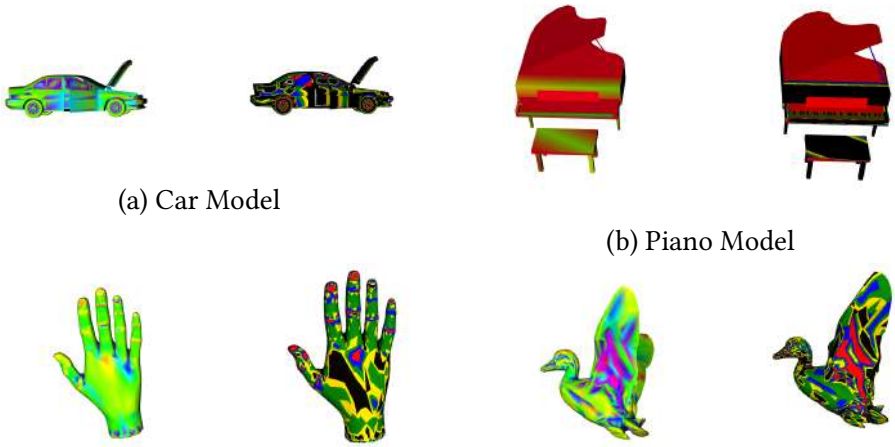
To maximize Q, first, each node in the network is assigned to its own community. Then, for each node i, the change in modularity is calculated to remove i from its own community and move it into the community of each neighbor j of i.

This process is easily evaluated in two steps:

1. removing i from its original community, and
2. inserting i to the community of j.

Finally, vertices that are similar in their geometric properties will have the same label. This stage will assess us to find correspondence between the regions of the two models. We provide in images 5.6 the descriptor map and region detection of different 3d models.

At the same time, we are providing the computational time with the respective number of vertices for several 3D models. It is obvious how the increase in the number of vertices consumes more time for clustering using the Louvain method. This is shown in table 5.1.



(a) Car Model

(b) Piano Model

(c) Hand Model

(d) Bird Model

Figure 5.6: Both Descriptor Map, along with their Feature Region Clustering for several models

Model Name	***Num. of Vertices***	***Time in sec.***
Piano	362	0.025
Hand	2489	0.94
Bird	2497	1.77
Car	5244	4.47
Cup	15209	149.63

Table 5.1: Time required for computation of the clusters in seconds

5.2.3 3D Surface Matching

After obtaining the feature regions, we represent this region by the centroid of these points that belong to the same region. So, for example, for such a model we get eleven feature regions; this means that we have eleven points that will be matched with those obtained from another model.

In another word, the matching is done by finding a relation between the maximum number of correspondences from the feature regions extracted. This problem is modeled as a linear sum assignment known as minimum weight matching. Formally, let X be a boolean matrix where $\chi[i,j] = 1$ iff row i is assigned to column j . Then the optimal assignment has cost.

$$min \sum C_{i,j}\chi_{i,j} \tag{5.7}$$

This problem is a modified Jonker-Volgenant algorithm without initialization, this algorithm is considered of $O(n)$ complexity as specified by [Vol89], and the al-

gorithm is widely described in [Cro16].

5.3 Experimental data analysis and results

In order to show the effectiveness of our proposed approach, we performed several experiments on free-download datasets from "MeshsegBenchmark" [CGF09], which was used for several algorithms, including [GF08] that explore a new form analytical technique based on randomized 3D surface mesh cuts. The main technique is to construct a random set of mesh segmentation and then count how many times each mesh edge intersects one of the randomized set's segmentation boundaries. The resulting "partition function" based on edges gives a continuous measure of where natural part boundaries appear in a mesh, while the collection of "most consistent cuts" gives a stable list of global shape attributes. Figure 5.7 shows their results for the segmentation. In addition, we used "CAPOD" data set [Pap14] which was offered

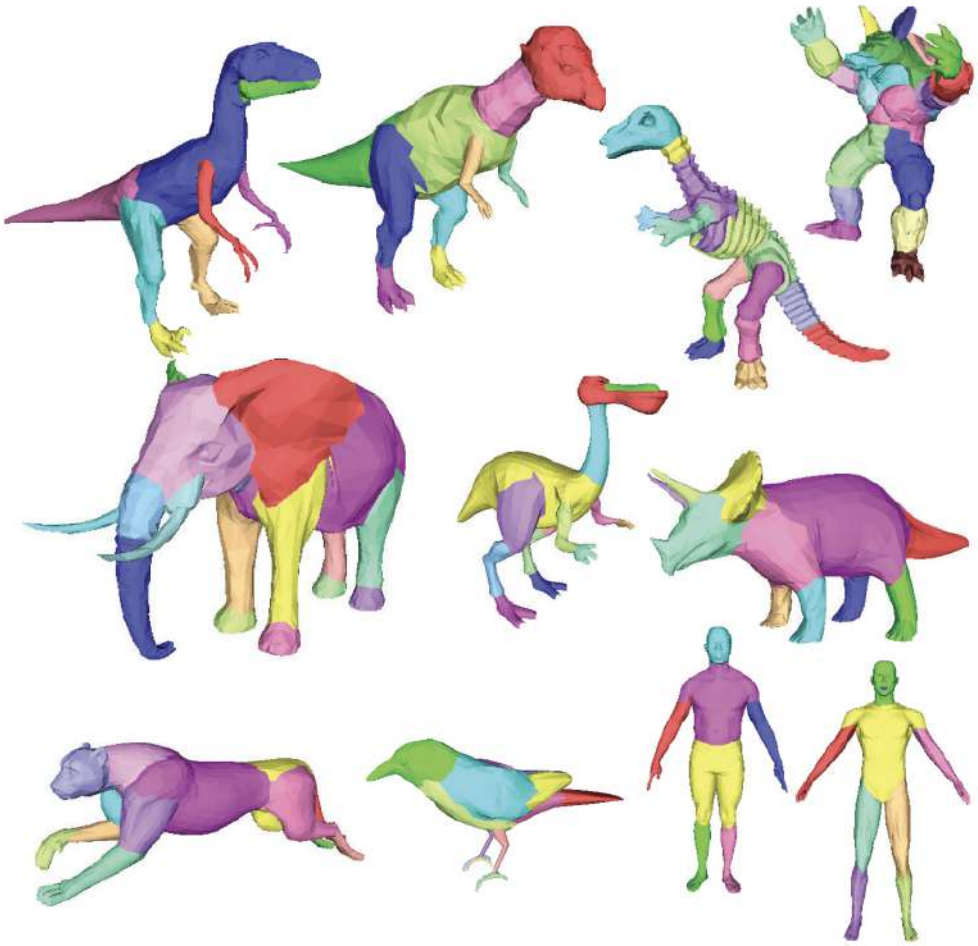


Figure 5.7: [GF08] Segmentation Results

to be the first 3D object ground-truth data set that permits objective evaluation of ap-

proaches for obtaining the canonical posture of objects in extrinsic space. 3D objects of the same class share a fixed pose in terms of object center, scale, and rotation due to the procedure used to compile the data set, while suffering various shape deformations. and the last data set was "Deformable Models" [SP04] where a full body key

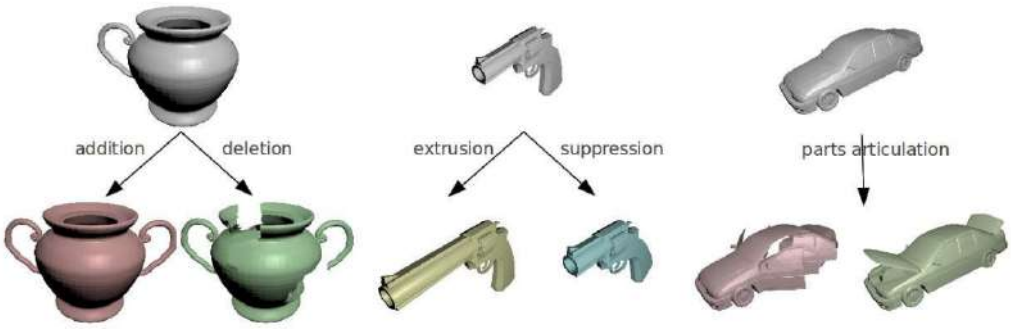


Figure 5.8: [Pap14] shape deformations used to produce object class members

poses are retargeted, scanned facial deformations are applied to a digital figure, and rigid and non-rigid animation sequences are remapped from one mesh to another.An example of their data set is found in Figure 5.9

Figure 5.9: [SP04] Facial expressions replicated from scans onto a digital character

The models are more than 500 objects, with different classes, each containing different poses.

Our proposition gives well matching despite the movement changes in human, teddy, armadillo, and the other models. Figure 5.10 shows the matching between different poses of the person; this obviously shows the invariance of our descriptor regarding movements.

The diversity between models within a class and the matching is seen in Figure 5.11, each sub-figure shows two matched models. The results ensure that our ap-

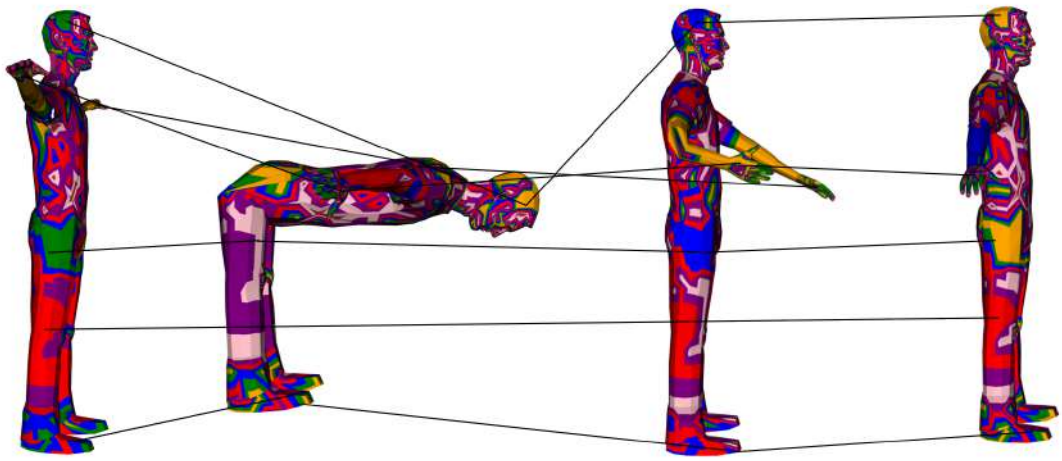


Figure 5.10: Matched Correspondences

proach is invariant to different affine transformations, whether rotation, scaling, or different poses.

3D Objects of the same class share a fixed pose in terms of object center, scale, and rotation while undergoing diverse shape deformations [Pap14], the matching between two of the same class with their feature regions visualization is presented in the following figure 5.12.

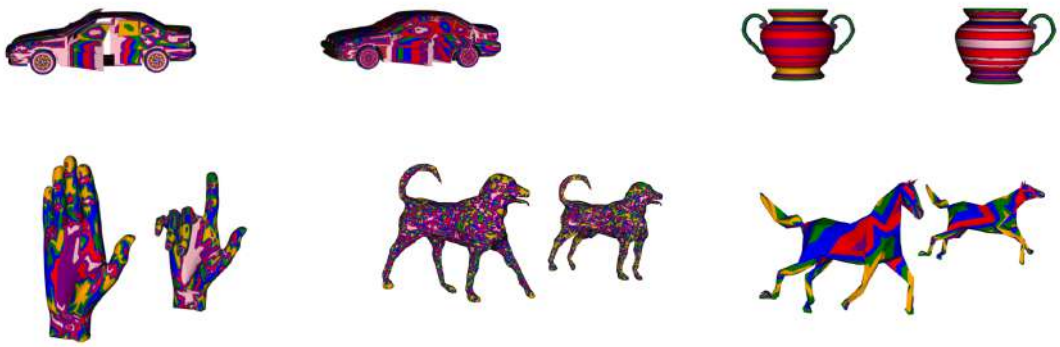


Figure 5.12: 3D matching within the same classes

Getting optimal matches encourages us to test our framework in order to classify the models, with a query model we try to retrieve the similar models with a cosine similarity of more than 80%, using the cosine similarity refers to the concept of, if the two similar models are far apart by the Euclidean distance (due to the size of the

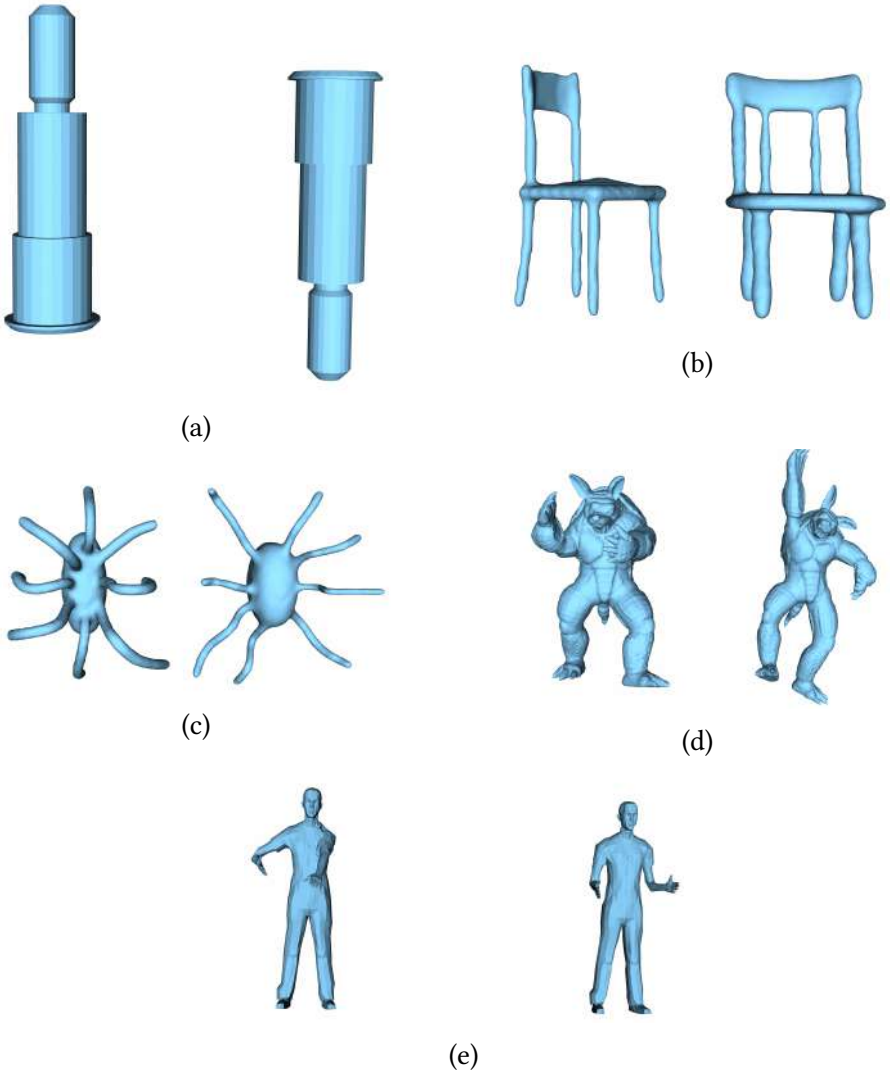
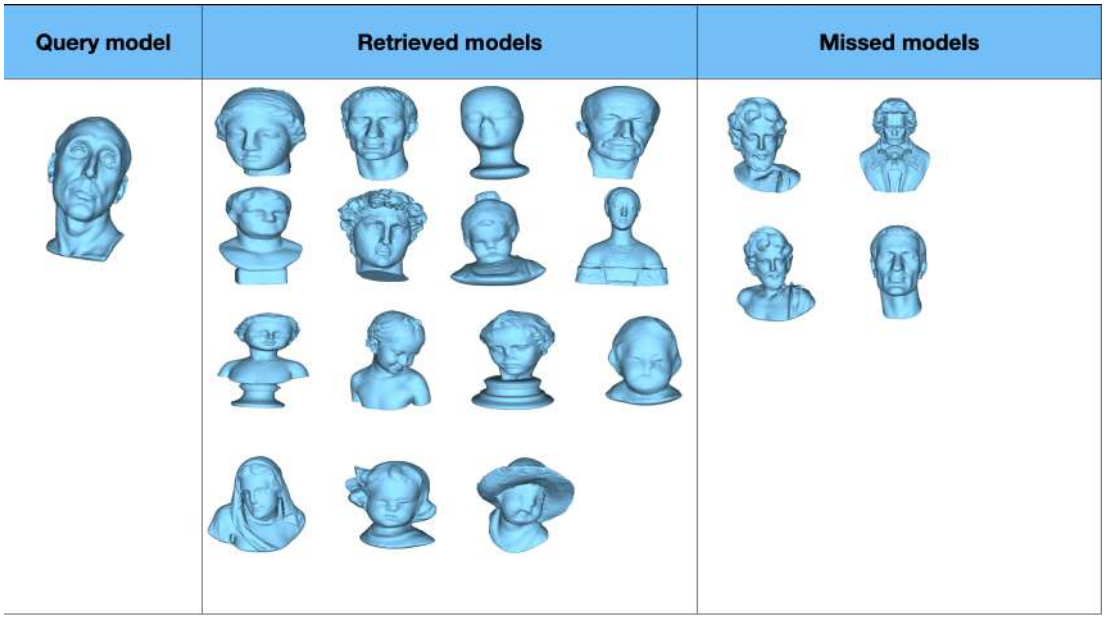


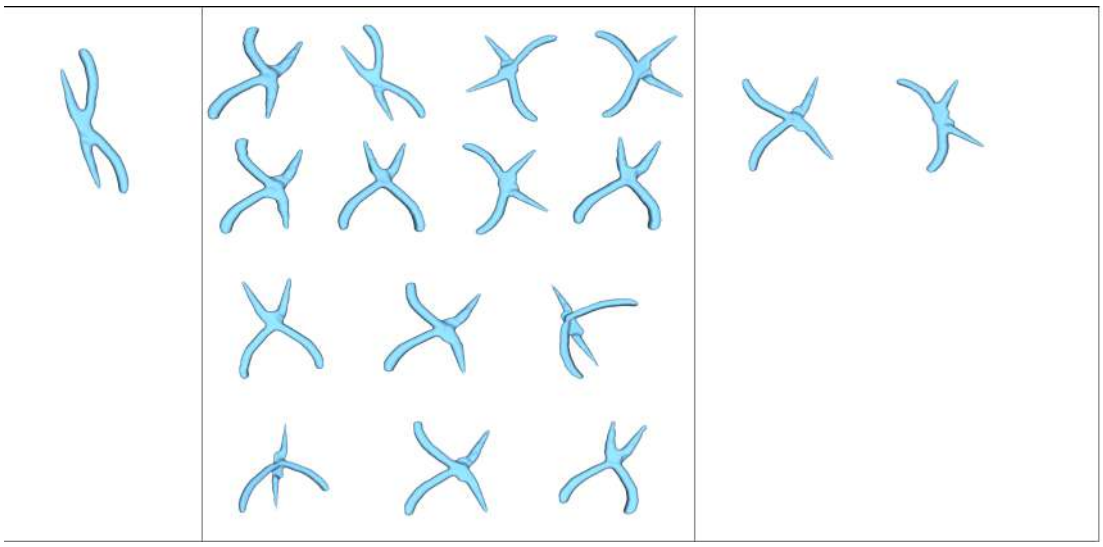
Figure 5.11: Articulated 3D shape matching

model), the chances are that they may still be oriented closer together through the angle between the models.

The classification of two query models as instances, where for each one, the retrieved models that refer to the same class, as well as the missed objects from the same class are shown in figure 5.13.



(a) Retrieval for Bust Model



(b) Retrievals for Pliers models

Figure 5.13: Classification

It is shown that somewhere our methodology lacks to obtain full retrieval for all the results, and this is shown for the fourth object that looks much similar to that of the query in 5.13a, as well to 5.13b where we missed two objects from having more than 80%.

Accuracy is the ratio of correct predictions to the total number of predictions. It is one of the simplest measures of a model. We must aim for high accuracy for our model. If a model has high accuracy, we can infer that the model makes the correct

predictions most of the time. And this is what we obtain through our framework, where almost every query gives more than 85% of accuracy. At the same time, we were able to get a high recall as a second measure.

5.4 Conclusion

Our work done in [SCA+22] showed promising results and took a step forward in finding a well-representative descriptor; this was a motivation to add more features in our descriptor with preservation, not to introduce bad complexity by this addition.

Those features extracted were used after following the concept suggested by [ECESAB20] about feature region detection. The instability in using meta-heuristic algorithms in their work led us to use the Jonker-Volgenant algorithm for solving the matching problem for the optimization problem modeling.

We compared our work to theirs, and were very satisfied. This was the motivation for including this contribution in an oral presentation at [SEA+]. As a summary for our methodology used in this chapter, we present a robust approach for 3D shape matching. The 3D shapes that were initially represented as meshes, and their surfaces were modeled as weighted graphs. A new combination of curvatures was extracted formulating a compact feature vector in order to cluster nodes, depending on the similarity measure.

Combinatorial optimization is used for the graph matching problem, this is done through the integer linear programming approach. Following this, we show the robustness and efficiency of the proposed approach, with experimental results applied to three different datasets with different affine transformations.

Finally, the classification results are presented, ensuring a wide range of applications to test. We believe that more experiments should be performed to study the gap in the false models retrieved.

Chapter 6

Brain Tumor Segmentation Based on α - Expansion Graph Cut

" In mathematics, you don't understand things. You just get used to them. "

– Johann von Neumann

6.1 Introduction 67
6.2 Related Work 69
6.3 Methodology 72
6.3.1 Materials 72
6.3.2 Methods 73
6.4 Implementation 74
6.4.1 Configuration and Preprocessing Step 75
6.4.2 Alpha-expansion graph cut 75
6.5 Results and Discussion 76
6.5.1 Pre-Processing Step 76
6.5.2 Segmentation 76
6.5.3 Performance Analysis 77
6.6 Conclusion and Future Work 83

Summary

This chapter introduces our work related to graph-cut in the field of medical imaging processing, we seek the segmentation of the brain tumor automatically without the user interaction.

6.1 Introduction

Through quantitative and computational techniques, medical image analysis plays an important role in biomedical sciences for studying, analyzing, and deciphering problems from medical imaging datasets acquired by various medical imaging modalities (such as X-Ray, CT-scan, MRI, and ultrasound).

MRI (Magnetic Resonance Imaging) is the most popular technique among these modalities. These quantitative image analysis techniques assist clinicians and medical experts in extracting important biological information from images for clinical decision-making, developing potential therapeutic strategies, and, most notably, neuroscience research.

Accurate and reliable brain tumor segmentation is crucial in these medical imaging modalities for undertaking safe diagnoses, healthy treatment planning, and consistent treatment outcome evaluation in order to understand and cure the complexities of chronic diseases such as cancer.

A brain tumor is an abnormal growth of tissue in the brain or central spine that can disrupt normal brain function and lead to a rise in brain pressure. Because of this, the tissues are pushed and this leads to the damage of the nerves of the other healthy brain tissues [KBA+07].

Segmenting the brain in general and whether to obtain a tumor or not, is very important in order to take further action, in addition to the case of having a tumor, to be able to further classify it into one of its three types.

Scientists had classified brain tumors based on:

1. How aggressive it is; its type and grade,
2. If it is cancerous (malignant) or not (benign), and
3. Where the tumor is located [BBO+07]

In addition, the cancerous tumor has several grades, we show them in the below figure 6.1, the first two tumors are considered a malignant tumor, contrary to the last one which represents a non-cancerous tumor.

In this part of the research, we focus on the type of segmentation which is the graph-based method, which we will give a more detailed view on it in the following.

A graph converts pixels or regions of the input image into graph nodes. The seg-

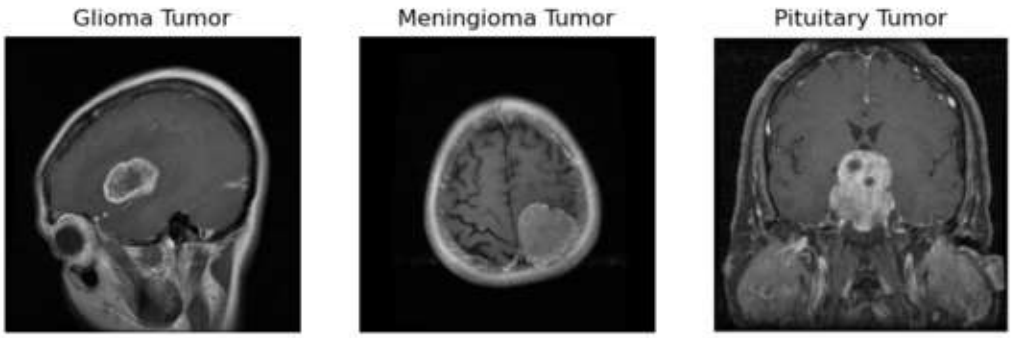


Figure 6.1: Tumor Types

mentation problem can then be transformed into a labeling problem, which requires assigning the appropriate label to each node based on its properties.

A graph G = (V, E) is a general structure that consists of a set of nodes (or vertices) V that correspond to pixels/voxels in the original image and a set of arcs (or edges) E that connect neighboring nodes. Every arc has a non-negative weight or cost that represents a type of quantity measurement based on the property of two neighboring vertexes connected by an edge.

Victor Chen and Su Ruan had proposed [CR09] an efficient method based on a generalized eigenvalue treatment to optimize the measurement of both, total dissimilarity among groups and total similarity within them. In their work, they lack a huge number of MRIs so it could not be well efficient specifically in the case of so many complicated MRIs.

According to their point of view [SM00, CBS05], they believe that the usage of graph-cut depends on the ability to segment the graph based on spectral analysis.

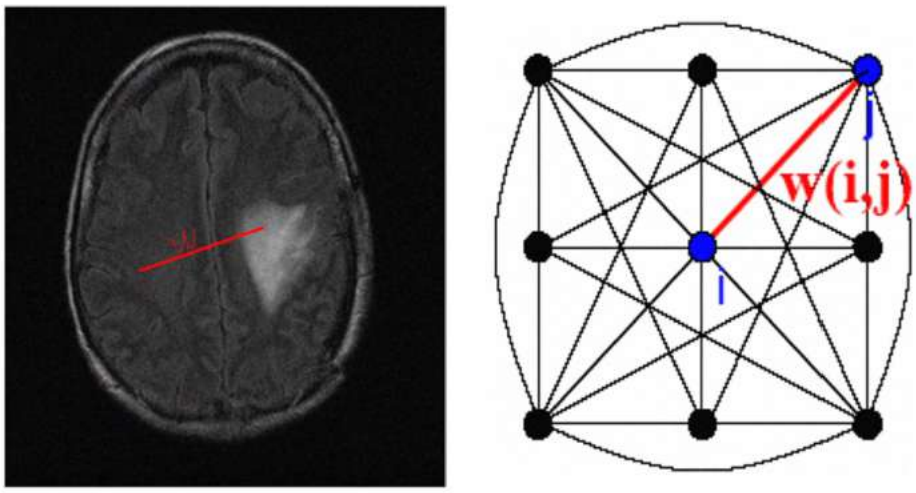


Figure 6.2: Referenced Image from [CR10]

Figure 6.2 illustrates the weight that represents the likelihood between pixels i and j inside a neighborhood and the map of connection design. During the decomposition procedure, they split the graph G into two sub-graphs A and B by separating the edges connected into portions. The total sum criterion can be used to compute this splitting method.

In [B.L19] a segmentation using grab cut algorithm was proposed, where they apply it by hard segmentation and border matting. The algorithm shows effective results except in the case of some sensitive brain structures where the accuracy of the conducted tumor was low.

Another work was done by Mayala et al. [MHH+22] by the use of a Minimum spanning tree(MST), where an interactive segmentation was proposed without tuning parameters, the author had performed a prepossessing step followed by the construction of the MST, then an interactive process by determining the background and region of interest(ROI) was required to complete the segmentation. Their method works for images with weak boundaries between the region of interest and the background, which requires more interactive steps to segment the tumor, which in our opinion is not feasible enough.

In the following section a more detailed related work is provided to enhance the importance of this research.

6.2 Related Work

Different techniques were used for the segmentation process. Most researchers went to classify image segmentation as broad topic into: threshold-based, pixel-based,

model-based, region-based, boundary-based.

The premise behind threshold-based approaches is that pixels that fall within a specific range belong to a particular class. The premise behind boundary-based approaches is that the pixel's characteristics change significantly where two sections converge. The assumption behind region-based approaches is that an area is made up of nearby pixels with comparable characteristics. In pixel classification algorithms, segmentation is based on feature space, employing pixel attributes, which might include the gray level, local texture, and color components for each pixel in the image. In model-based procedures, such as in, a model is created by incorporating prior knowledge about an item or specific anatomical structure, such as its shape, size, texture, and orientation. Some hybrid strategies combine two or more of the aforementioned techniques [CV01, VA15, BRT17, Ots79].

In order to show the importance of this work, the huge research done for it, we include in the figure 6.3, in addition to 6.4 that shows partial image retrieving the number of surveys from 2012 till 2019 and number of methods compared in them, both obtained from [HM19].

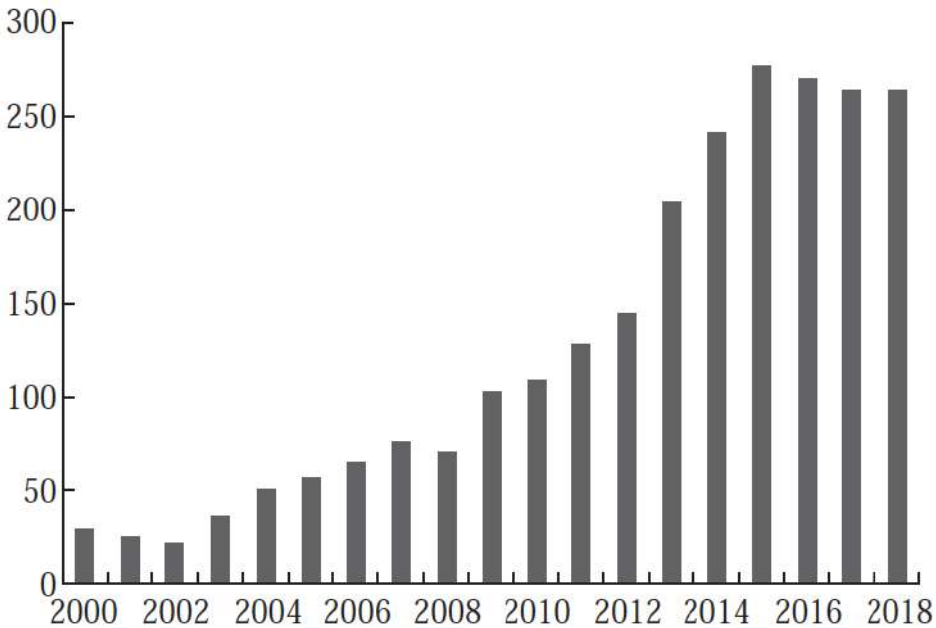


Figure 6.3: Number of papers on brain tumor segmentation approaches from 200 till 2018, obtained from [HM19]

As specific work for segmenting MRI to obtain brain tumor, Hussain et. al in their work [HAM17] uses deep convolutional neural networks (DCNN) to perform brain tumor segmentation. They fixes the over-fitting of DCNN by the use of max-out and drop-out. In their algorithm, they used both pre-processing and post-processing to get rid of small false positives results.

Year	Number of methods reviewed in each survey papers
2012	44
2012	34
2013	55
2013	39
2014	31
2014	49
2015	32
2018	45
2018	31
2018	36
2018	61
2019	72

Figure 6.4: Partially cropped image from [HM19] showing the surveys done from 2012 till 2019 with the number of methods used in each

In the same region of working, neural networks, the study done by [SAK+18], follows the concept of Ensemble Learning to design a more efficient model, and in order to accelerate the training process they tend to influence hyper-parameters by bounding and setting a roof to these hyper-parameters.

Nevertheless, and due to the following reasons:

1. A large amount of data are needed for CNN,
2. A convolutional neural network takes a very long time to train, especially with huge datasets. To speed up training, you typically need specialized hardware (like a GPU).
3. Because of operations like maxpool, CNNs often operate significantly more slowly.

We search for another type of segmentation, in order to be more flexible, regarding whether the hospital has not enough MRI images to process, as well the key concept behind the automatic segmentation is to fasten and ease the process for physicians and doctors.

Since we believe that the use of threshold-based methods can be used only as pre-procesiing step, it is not well efficiently to segment tumor alone, and since our main focus in research is the graph structure, we tend to study the graph-based approaches for segmenting tumor.

The study done by Parisot et al. in [PDCP12] tends to make use of previously acquired information in the form of a sparse graph that depicts the anticipated geographical placements of tumor classifications. Such data is combined with image-based classification methods, spatial smoothness restrictions, and detection map reliability requirements to produce a consistent graphical model formulation.While their work seems promising, they lacks a high registration errors.

According to [AP19], to segment a brain tumor from multi-modal MRI images, a new Walsh Hadamard Transform (WHT) texture for super-pixel-based spectral clustering is proposed. To produce more exact texture-based super-pixels, the Simple Linear Iterative Clustering (SLIC) technique is first used to create texture saliency maps using the chosen WHT kernels. Then, for segmenting brain tumors in MRI images, the texture super-pixels become nodes in the graph of spectral clustering.

The huge open research in graph-based segmentation methods that shows up lately, encouraging us to search for optimal graph algorithm, that is optimal in complexity, run time, and has an efficient segmentation results.

6.3 Methodology

Our proposed approach implements an automated setup to mine the fatal region in 2D brain MRI. It is based on the energy minimization for graph-cut. We will define some basic notation that is applied to graph cuts in the context of our segmentation method.

6.3.1 Materials

In this work, two datasets were used; The first one is obtained from Kaggle [Cha19], and it was targeted to make segmentation without the numerical results due to the unavailability of the ground truth of these MRIs.

The second data set is referenced by [Che17]. This data set and as mentioned in their work [CHC+15] was obtained from Nanfang Hospital, Guangzhou, China, and General Hospital, Tianjing Medical University, China, from 2005 to 2010. It contains 3064 slices from 233 patients and is classified into three labels: 1) glioma, 2)meningioma, and 3) pituitary tumors. As well the slice thickness is 6 mm and the slice gap is 1 mm. We refer the reader to their main work [CHC+15, CYH+16] for checking the original images.

It is important to mention that the data provided are available in Matlab form (.mat). Each file stores a structure that contains different fields for an image. The fields included in each file are labels for tumor type, patient ID, image data, tumor borders, and tumor masks.

6.3.2 Methods

This section is started by defining some principal terms that considered important before proposing the approach.

Graphs in Computer Vision

In computer vision, an image is typically represented as a graph with each pixel or super-pixel acting as a vertex and being connected to other defined neighbors (the most common is 4, up, down, left, right, and it is also fully connected).

It can be formulated as an energy minimization problem (with a coarse-to-fine converging method or a graph cut in an unsupervised image segmentation task) or as a probabilistic graphical model depending on the task (which maximizes the probability).

And these two types could eventually lead to the same problem: optimization.

In our approach, the 2D sliced MRI is mapped onto a weighted graph, with two distinct vertices called terminals. Let $G = (V, \xi, \omega)$ be a weighted graph with V the set of vertices and ξ the set of edges connecting every two vertices with a ω for the weight on their edges. In many works related to graph-cut, the weight represents the cost between nodes c.

A cut $C \subset \xi$ is a set of edges such that the terminals are separated in the induced graph $G(C) = (V, \xi - C)$. In addition, no proper subset of C separates the terminals in $G(C)$. The cost of the cut C, denoted $|C|$, equals the sum of its edge weights.

The minimum-cut problem is to find the cut with the minimum cost. Many algorithms were formulated in order to solve this low-order polynomial complexity [AMO93]. The claim that it is polynomial complexity is very important because of the consequences of treating large graphs. The following subsection will show the energy minimization as well as the graph-cut algorithm method to be used in our work.

Graph cut using Energy minimization algorithms: alpha-expansion

As what we will be mentioning in the implementation part, we will be dealing with a binary problem to figure out whether the MRI where has a tumor or where not. So, Graph Cuts can be used to find the optimal solution for this binary problem.

The type of energy here is addressed by Greig et al. in his work [GPS89], in addition to most of the graph-based methods including [IG98, BVZ98, RG00, SVZ00, Vek00, TBRN00] but not limited to.

$$E(L) = \sum_{p \in \mathcal{P}} D_p(L_p) + \sum_{(p,q) \in \mathcal{N}} V_{p,q}(L_p, L_q) \tag{6.1}$$

The best cut is obtained by minimizing the above energy function equation 6.1 which leads to optimal segmentation for our MRI. This equation has two terms; the first represents the data term where $\mathcal{P}$ is the pixels in our image, and L represents its labeling, in addition to the second term that represents binary interactions between vertices where $\mathcal{N}$ is a set of all pairs of neighboring pixels.

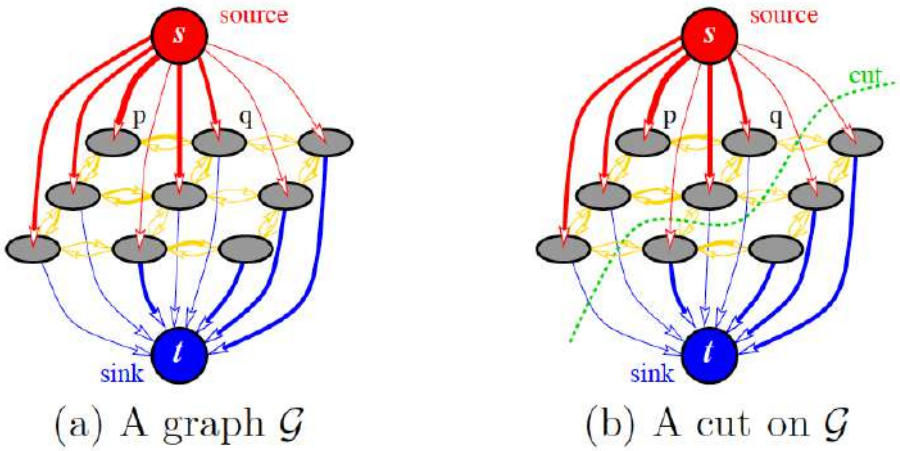


Figure 6.5: A directed capacitated graph, with costs of edges, is reflected by their thickness. A similar graph-cut construction was first used in vision by Greig et al. [GPS89] for a binary image. restoration.

Figure 6.5 (a) is a simple example of a 3 x 3 image with a source s and sink t; at the same time, it is a simple consequence of the fact that normal graph nodes represent regular image pixels. Each edge connecting two vertices is assigned some cost(weight). An s/t cut in a graph G is a partitioning of V into two disjoint subsets S and T such that all the image pixels are connected to an object terminal node s and all the background pixels are connected to a background terminal node t, which is shown in Figure 6.5 (b).

After building a graph, a pseudo-Boolean function is represented, and it is possible to compute a minimum cut using one of the various algorithms developed for flow networks, such as the Edmonds-Karp [EK72], Ford-Fulkerson [FF56], and Boykov-Kolmogorov algorithm [BVZ98]. Due to its efficiency and many improvements applied to it, we tend to use the enhanced algorithms done by Boykov et al. in their work [BVZ01].

6.4 Implementation

The experiments done through this work were performed on MacBook Pro Apple M1 8Gb with the use of Python [VRDJ95] as a programming language and PyCharm as an IDE.

6.4.1 Configuration and Preprocessing Step

As mentioned in the materials used 6.3.1 the first data set is optimal to start working with directly, while regarding the second data set, we need to prepare it where we obtain the MRIs and the masks from the file and obtain a .jpg for each, which are the input for our algorithm. By the use of [Col13] and [Cla15] we were able to obtain a proper dataset as .jpg MRIs.

As a pre-processing step, we needed to only:

1. Convert image into a greyscale image.
2. Apply a Gaussian filter to reduce or remove artifacts with a 5x5 kernel.

Testing images for the pre-processing step and the full results are shown in sections later.

6.4.2 Alpha-expansion graph cut

The core idea of this algorithm is to successively segment all α and non-α pixels with graph cuts and change the value of α at each iteration. The algorithm will iterate through each possible label for α until it converges.

At each iteration, the α region $\mathcal{P}_\alpha$ can only expand. This will change the way to set the graph weights. Also when two neighboring nodes do not currently have the same label, an intermediate node is inserted and links are weighted so they are relative to the distance to the α label.

This algorithm was proven to be faster than Alpha-Beta Swap Algorithm. The pseudo-code of the algorithm shown in 1, with the representation of the graph in figure 6.6.

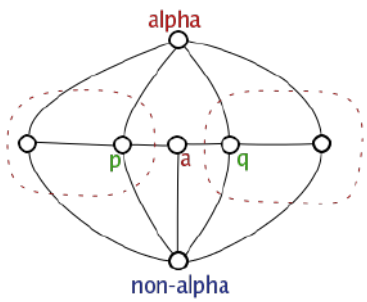


Figure 6.6: Alpha-Expansion Graph

Algorithm 1 Expansion Move Algorithm

1. Start with an arbitrary labeling
2. Cycle through every label α
   - 2..1 Find the lowest E labeling within a single α - expansion
   - 2..2 Go there if it's lower E than the current labeling
3. If E did not decrease in the cycle, done Otherwise, go to step 2

Figure 6.7 shows how the segmentation process which is done for the α expan-

sion algorithm, where the original image is 3x3, each pixel was represented as a node in the graph.

Adding two external nodes where while segmentation, each segment will refer to a node, foreground, and background.

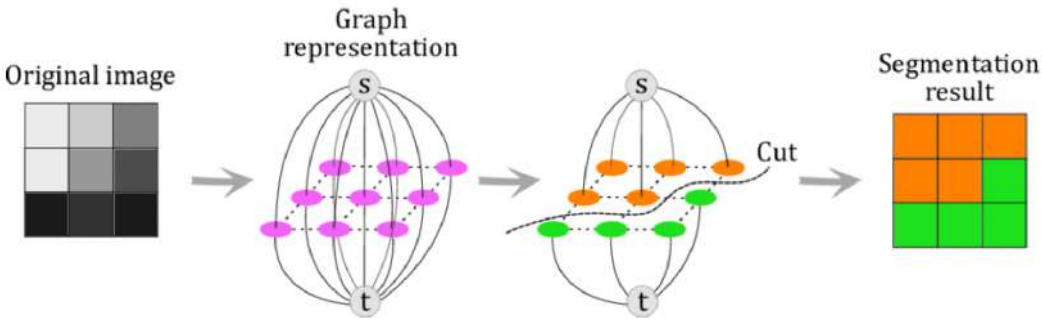


Figure 6.7: Steps from: Original Image towards Segmentation

As a final step for the segmentation, post-processing is applied in order to obtain the large contour where depending on the experiments shows that it refers to the skull, we need to eliminate it.

Section below, we will show several experiments for the two datasets and discuss the results.

6.5 Results and Discussion

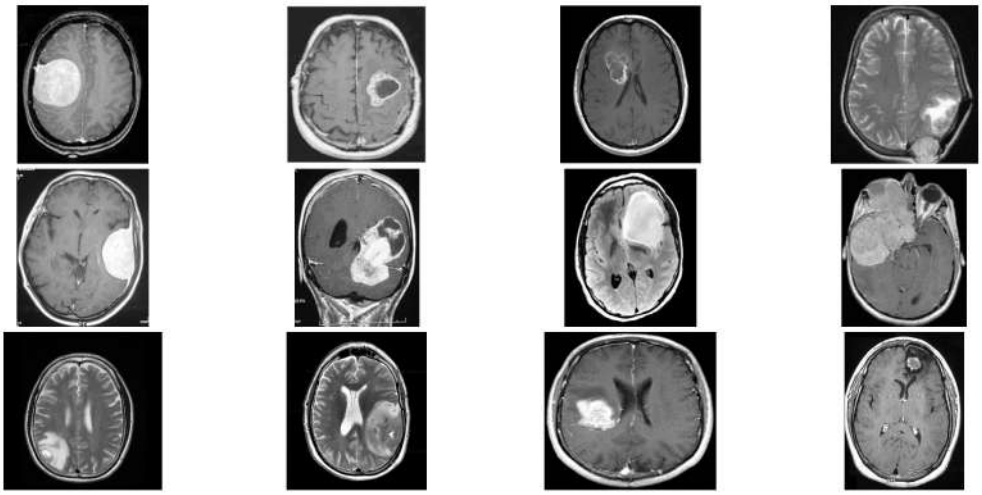
6.5.1 Pre-Processing Step

As mentioned earlier this step plays an important role to prepare our work for the segmentation. It clean the images from artifacts and noise, in addition to the fact that by getting a grey scale image, we are working with 2D instead of 3D arrays.

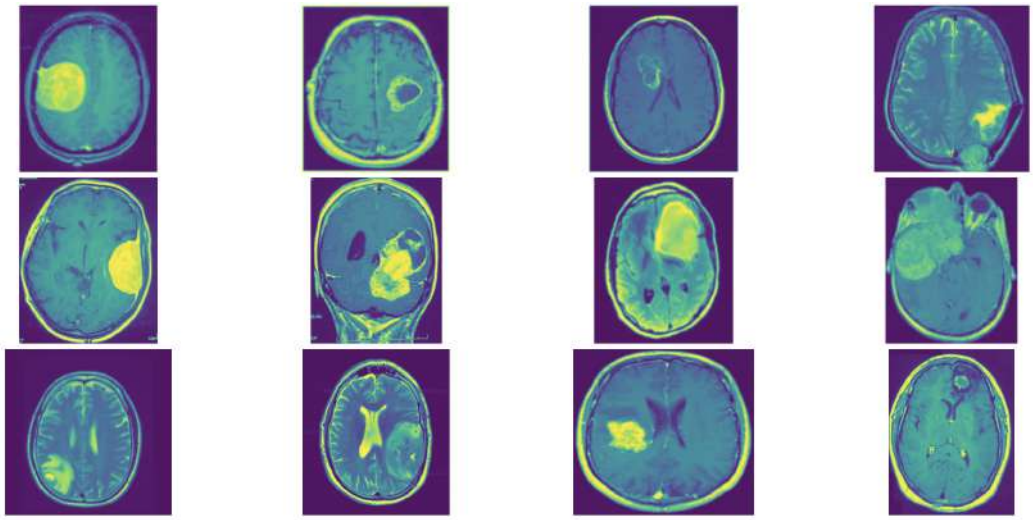
Ten samples are shown in figure 6.8, the initial as well the greyscale filtered images.

6.5.2 Segmentation

We are showing in Figure 6.9 examples for the pre-processed image, how it is segmented by the α-expansion method and then by the post-processing how we obtain the final tumor region, and the last row represents the ground truth.



(a) Initial Images



(b) Grey Filtered Images

Figure 6.8: Initial Images with their greyscale filtered images

6.5.3 Performance Analysis

Ensuring the efficiency of our results numerical tests were performed. The most popular metrics are accuracy, specificity, and precision, where these values range between zero and one, where one means there is a perfect overlap between ground truth segmentation and the one obtained using the proposed method, while zero means no overlapping.

To apply these metrics we tend to binaries our 2D MRI segmented image, and that for ground truth. We will have a label for each image, L_G refers to the ground truth, while L_S refers to our approach.

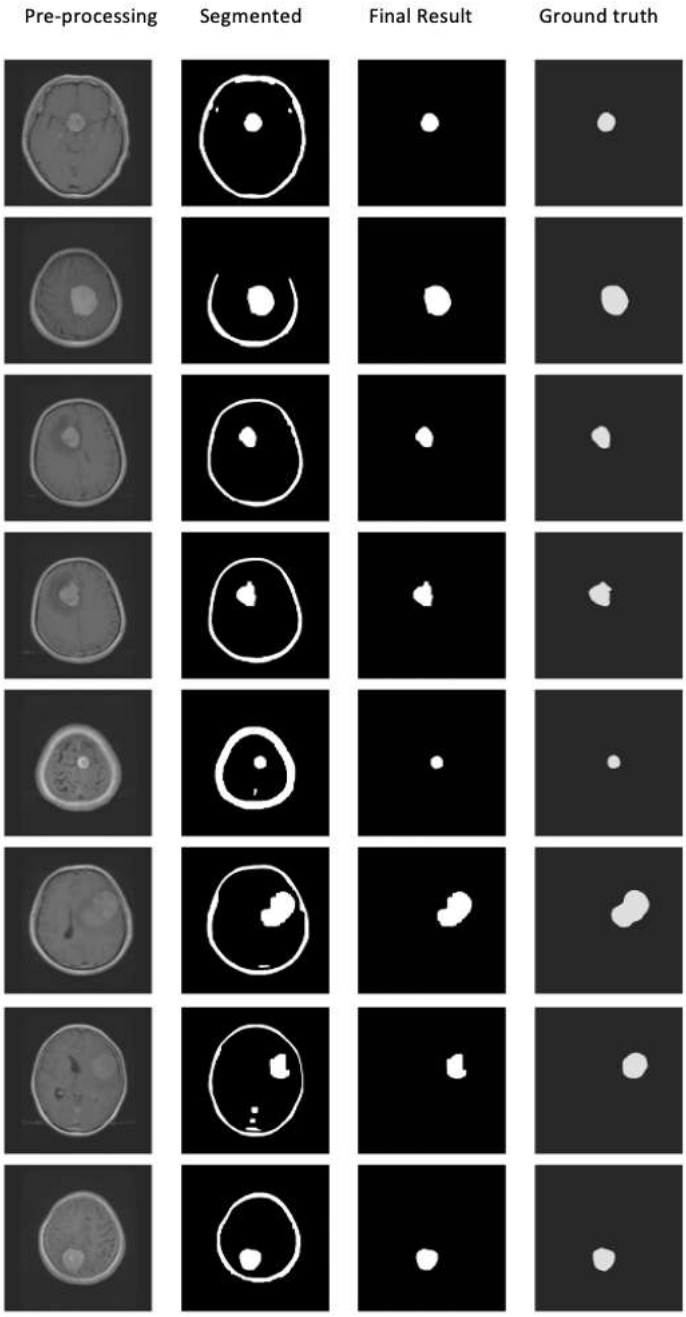


Figure 6.9: The full approach compared to the ground truth of eight samples

The concept of true positive (TP), false positive (FP), true negative (TN), and false-negative (FN) is to check the performance of the method. TP represents the labels that are correctly classified as a brain tumor. FP represents the labels that are incorrectly classified as a brain tumor.

They are not in the tumor region but are classified as being in a brain tumor

region. TN are labels that are correctly classified as non-tumor material, FN represents the labels that are incorrectly classified as non-tumor materials. The concept is paraphrased from [HHG+20].

For the Specificity(Sp), Accuracy (A), and Precision (P), Jaccard Index (JI), Dice Similarity Coefficient (DSC), and Sensitivity (Sn):

$$Sp = \frac{TN}{TN + FP}, \tag{6.2}$$

$$A = \frac{TP + TN}{TP + TN + FN + FP}, \tag{6.3}$$

$$P = \frac{TP}{TP + FP}, \tag{6.4}$$

$$JI = \frac{TP}{TP + FP + FN}, \tag{6.5}$$

$$DSC = \frac{2TP}{2TP + FP + FN}, \tag{6.6}$$

$$Sn = \frac{TP}{TP + FN} \tag{6.7}$$

The numerical results are shown for ten samples, two more than that included in the figure 6.10.

Our results in most of the samples shows high numerical results due to its efficiency. The values ranges between 0 and 1. A value zero means there is no overlap between the segmented region using the proposed method and the ground truth. The value 1 means there is a perfect overlap between ground truth segmentation and the one obtained using the proposed method.

Almost in all results we obtained the Precision, Sensitivity, specificity, and accuracy are almost nearly 1, but badly in few cases the jaccard indicies and precision tends to be lower than 0.8 and we believe that this refers to the complex structure of the brain.

We shows the images in the figure below that lacked getting high precision, and jaccard index.

Comparison with others work

The work done by [B.L19], tend to apply the grap-cut technique. By resolving the min cut based graph problem, it is possible to extract the image segmentation into the background and foreground regions. When segmentation is done manually, regions are identified and assigned to either the sink or source node. The segmentation process uses the color information provided by the chosen regions to define the

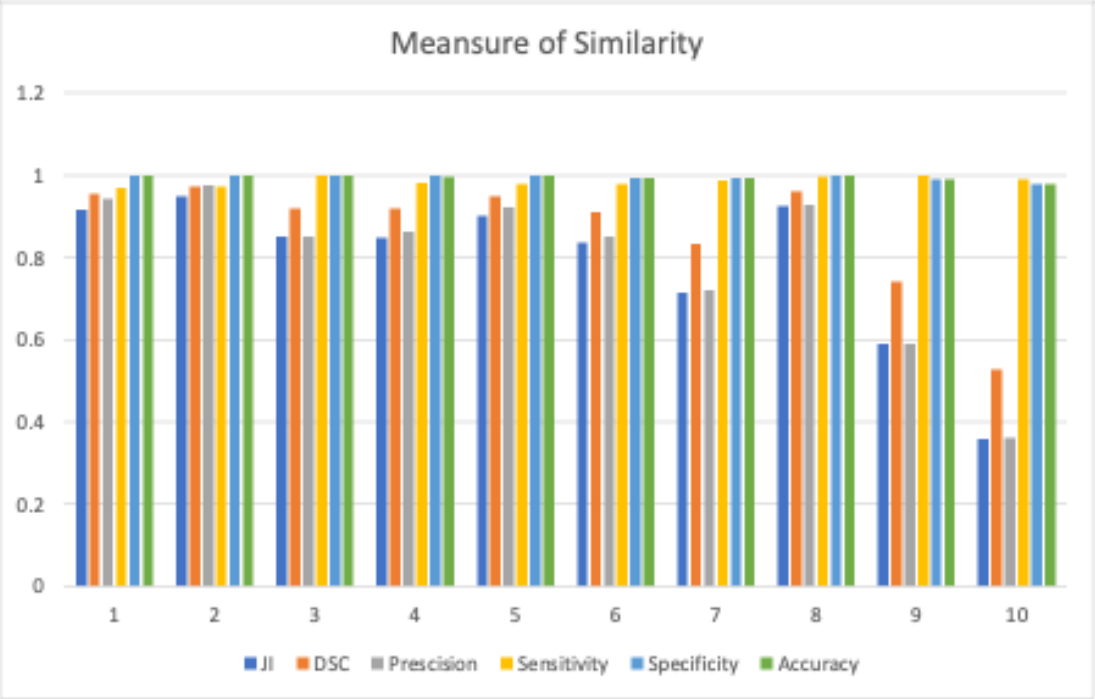


Figure 6.10: Numerical Results for ten samples taken randomly.

background and the object.

The performance analysis for their method was shown only for five images, and it is ranging between 85% and 99% which shows on average around 93%.

For the precision parameter their work outperforms ours, where they shows that it tends to be more 95% while ours is about 80%.

The specificity in our work compared to the work stated in [MHH+22] are nearly the same. We believe that due to the usage of same data set used by the authors, our work is optimal and the comparison is more logic, in addition that in some cases the jaccard indicies are low, but still more than the half. We believe that this refers to the complicated structure of brain in some cases.

Images in the figures below represents a sampling from the two datasets we have, some have complex structure, others are better.

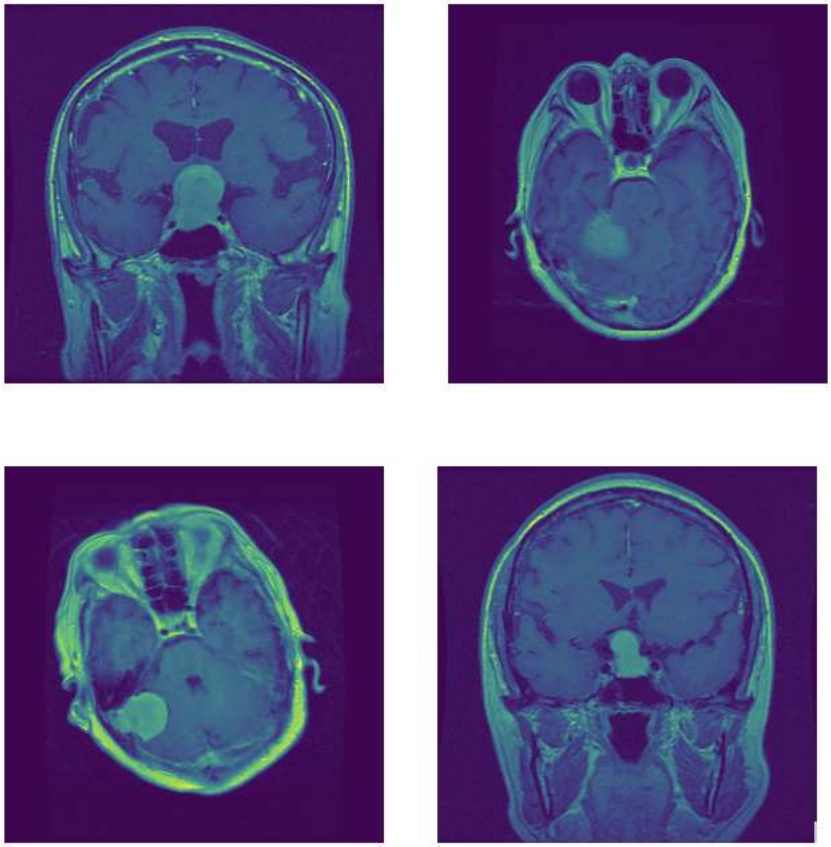


Figure 6.11: Images that failed in their full segmentation

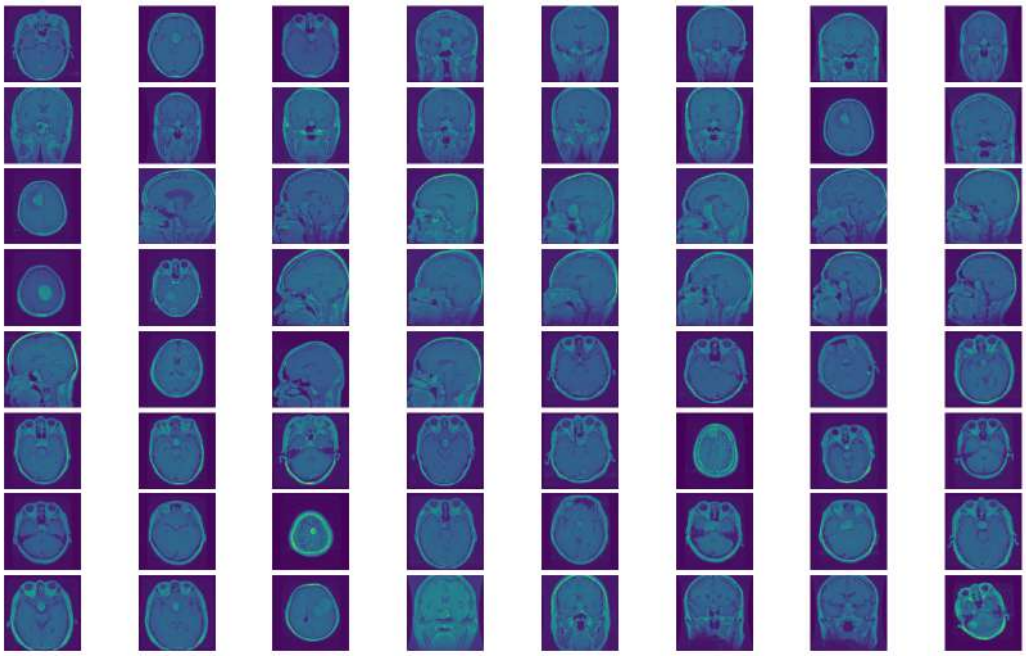


Figure 6.12: Samples that all of size 512x512 from [Che17]

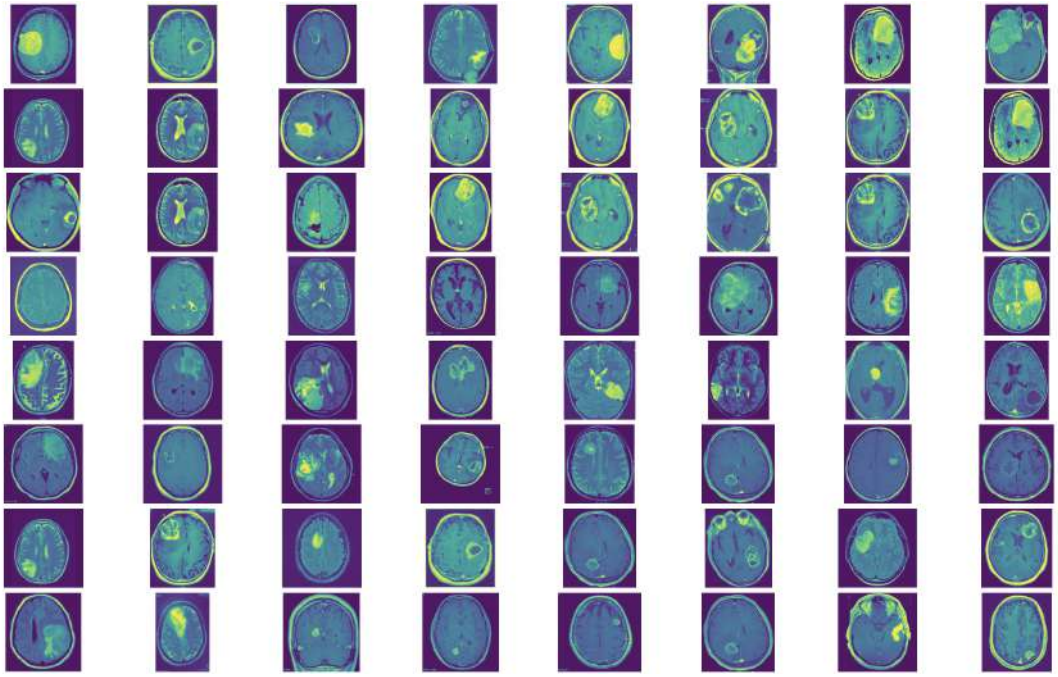


Figure 6.13: Samples that has different sizes from [Cha19]

6.6 Conclusion and Future Work

The use of the graph-cut method is an optimal way to segment an image. In this work, we propose an approach to detect brain tumors from 2D MRIs. This approach relays on preparing the MRI by pre-processing steps to be fed up to a powerful algorithm, α-expansion move graph-cut, followed by post-processing to finalize the segmentation. Our approach shows efficient percentages on the level of accuracy, specificity, and precision.

Each single step of our approach is considered a usual step in specific application, but our originality in this research is shown in the combination and how the flow of the algorithm is done.

Our algorithm was able to be applied to more than 400 MRIs in addition to good results with optimal time.

We tend for further work to perform experimental results on more complicated structures for the brain, comparing these results with another approach using the same data sets, in addition, to make this framework an initial step in classification the tumor to which grade it belongs.

Chapter 7

Conclusion

"Success is no accident. It is hard work, perseverance, learning, studying, sacrifice and most of all, love of what you are doing or learning to do."

– Pele

7.1 Conclusion 85
7.2 Future Perspective 86
7.3 French Conclusion 88
7.4 Perspectives 89

Summary

In this chapter we conclude the work of three years and a half, we also include some future perspectives to be done with more time and better circumstances. Also, a french conclusion is available.

7.1 Conclusion

In this dissertation, we developed two distinct approaches to 3D models: formulating a new combination of well-discriminating descriptors that allows us to match two 3D objects in a good matter and retrieving and classifying objects. Several graph methods were deployed, finding the optimal with better results and more efficient in time and complexity.

Our work in the first part 4 was dedicated to making a self-comparison between two approaches that solve matching problems: Hungarian and brute force, and this was done after ensuring the best fit between the two ways of computing the 3D Harris detectors. This work was the output of a long journey that required a lot of research as a newbie in the academic field.

This article was followed by a proceeding 5 with the following output: Simultaneously, instead of using the 3D Harris Detector, we tried another approach to simplifying our 3D models, where we tend to use the Louvain clustering method, in addition to computing the weights for the sub-graphs. This work was compared to that done by [ECESAB20], and our work showed optimal results and was well evaluated.

We found a good representation for the 3D models; it is a combination of effective descriptors that with their combination did not affect the complexity of obtaining them. As mentioned previously, it is used to combine two descriptors, but in our work, we combined three, and in another work, we combined four.

In this thesis, we raise the question of the better algorithm to find the optimal match, and depending on the experimental results, the Hungarian algorithm performs better than brute force. In addition, we applied the assignment problem that will be later compared to Hungarian to evaluate the best between them.

We showed in the second work how the effectiveness of clustering assists in lowering the size of the problem and consequently having good results for one-to-one correspondence.

The evaluation of the descriptors is well studied by checking for correct matches and well classification. The descriptor is considered discriminating when it is invariant to the affine transformations. Throughout the experimental results, our formulated descriptor, whether the first or the second, was able to match different models with different poses and with many affine transformations.

In the second part of our manuscript 6, since the medical imaging field is one of the most important fields nowadays, we seek to extend our research in that domain. Brain tumor-specific is one of the most common diseases afflicting people worldwide. Much research has been done before, and we seek to work on something nearly never used in this specific field.

Our main intention is to segment the tumor at an early stage, which leads to the beginning of treatment earlier. Several experiments were conducted to reach the final aim, but due to the inability to reach the targeted goal, we preferred not to mention them. We end up with an optimal framework for the segmentation of brain tumors based on a graph cut that has a good representation model.

In our framework, we provide a method to identify brain tumors in 2D MRIs. This method relies on preprocessing the MRI to feed it to the alpha-expansion move graph-cut algorithm, then postprocessing to complete the segmentation. Our method displays effective percentages for the degree of precision, specificity, and accuracy.

In some applications, every single step of our approach is seen as commonplace. However, the combination and execution of the algorithm flow are where our originality in this research is demonstrated.

In this thesis, we were able to answer the questions asked in the introduction section 1.2.

Moreover, our research to some degree was able to find a good approach for segmenting brain tumors even with the complicated structure of the brain, the low resolution in some cases for MRIs, and different factors. The work in the 2D images by the preprocessing step, specifically the filtering phase, gives better evaluation, which leads to cropping MRIs in a well-defined manner and better segmentation.

7.2 Future Perspective

Many ideas arise after each piece of research, but all of these require more time, more resources, and many other things. In the field of 3D, cases about applying multi-scale descriptors and their effectiveness are proposed and considered as one of the main focuses to continue working on them for a while.

At the same time, with the successful approach in Chapter 5, applying it to different real-world applications could be suggested as a master's thesis.

For the part related to medical imaging and future perspectives, we are proposing to make this segmentation a first step that comes before classifying the grade of the tumor by studying the features of the region and applying several recent methodologies in this domain related to meta-heuristic algorithms, Support Vector Machine (SVM), Linear Regression, and many other algorithms for classification that are promising.

In addition to deploying this as a desktop application to assist physicians and

doctors in hospitals, facilitating the segmentation and showing the relevant needed parameters regarding the area of the tumor, volume, grade, and other specifications are found.

In this conducted research, we tried to benefit from the powerful tool "graph representation" in two domains: three-dimensional and two-dimensional data. The mentioned future perspectives are suggestions that we are still working on them despite the end of the Ph.D. period having passed.

The work carried out in this thesis has revealed a publication in international journal of computational vision and robotics and can be found in [SCA+22].

In addition to an oral presentation in an international conference with a published work presented in [SEA+].

According to the last work related to brain tumor segmentation found in Chapter 6 we submitted it to the International Journal of Imaging Systems and Technology.

7.3 French Conclusion

Dans cette thèse, nous avons développé deux approches distinctes des modèles 3D :

1. Formuler une nouvelle combinaison de descripteurs bien discriminants qui nous permet de faire correspondre deux objets 3D dans une bonne matière et récupérer et classer des objets.

2. Développer plusieurs méthodes de graphes pour trouver l'optimum avec de meilleurs résultats et efficacement en termes de temps et de complexité.

Notre travail dans la première partie a été consacré à faire une auto-comparaison entre deux approches qui résolvent des problèmes de correspondance : "algorithme hongrois" et "algorithme force brute". Cela a été fait après être assuré du meilleur ajustement entre les deux façons de calculer les détecteurs Harris 3D. Ce travail est le résultat d'un travail sérieux qui a nécessité beaucoup de recherches en tant que débutant dans le domaine académique.

Ce travail a été suivi d'une procédure avec le résultat suivant : Simultanément, au lieu d'utiliser le détecteur Harris 3D, nous avons essayé une autre approche pour simplifier nos modèles 3D, où nous avons tendance à utiliser la méthode de regroupement de Louvain, en plus de calculer les poids pour le sous-graphes. Ce travail a été comparé à celui effectué par [ECESAB20], et notre travail a montré des résultats optimaux et a été bien évalué.

Nous avons trouvé une bonne représentation pour les modèles 3D. C'est une combinaison de descripteurs efficaces qui, avec leur combinaison, n'affecte pas la complexité de leur obtention. Comme mentionné précédemment, il est utilisé pour combiner deux descripteurs, mais dans notre travail, nous en avons combiné trois, et dans un autre travail, nous en avons combiné quatre.

Dans cette thèse, nous soulevons la question du meilleur algorithme pour trouver la meilleure correspondance, et selon les résultats expérimentaux, l'algorithme hongrois est plus performant que la force brute. De plus, nous avons appliqué le problème d'affectation qui sera plus tard comparé au hongrois pour évaluer le meilleur entre eux.

Nous avons montré dans le deuxième travail comment l'efficacité du clustering aide à réduire la taille du problème et par conséquent à avoir de bons résultats pour la correspondance biunivoque. L'évaluation des descripteurs est bien étudiée en vérifiant les correspondances correctes et la bonne classification. Le descripteur est considéré comme discriminant lorsqu'il est invariant aux transformations affines.

Tout au long des résultats expérimentaux, notre descripteur formulé, qu'il soit le premier ou le second, a pu faire correspondre différents modèles avec différentes poses et avec de nombreuses transformations affines. Dans la deuxième partie de

notre manuscrit, puisque le domaine de l'imagerie médicale est l'un des domaines les plus importants de nos jours, nous cherchons à étendre nos recherches dans ce domaine. Les tumeurs cérébrales spécifiques sont l'une des maladies les plus courantes qui affligent les gens dans le monde.

Beaucoup de recherches ont été faites auparavant, et nous cherchons à travailler sur quelque chose de presque jamais utilisé dans ce domaine spécifique. Notre intention principale est de segmenter la tumeur à un stade précoce, ce qui conduit à un début de traitement plus précoce. Plusieurs expériences ont été menées pour atteindre le but final, mais en raison de l'impossibilité d'atteindre le but visé, nous avons préféré ne pas les mentionner.

Nous nous retrouvons avec un cadre optimal pour la segmentation des tumeurs cérébrales basé sur une coupe de graphe qui a un bon modèle de représentation. Dans notre cadre, nous fournissons une méthode pour identifier les tumeurs cérébrales dans les IRM 2D.

Cette méthode repose sur le prétraitement de l'IRM pour l'alimenter à l'algorithme de coupe graphique (graph-cut algorithm) de déplacement alpha-expansion, puis sur le post-traitement pour terminer la segmentation.

Notre méthode affiche des pourcentages effectifs pour le degré de précision, de spécificité et d'exactitude. Dans certaines applications, chaque étape de notre approche est considérée comme courante. Cependant, la combinaison et l'exécution du flux de l'algorithme sont là où notre originalité dans cette recherche est démontrée. Dans cette thèse, nous avons pu répondre aux questions posées dans l'introduction section 1.2.

De plus, nos recherches ont pu dans une certaine mesure trouver une bonne approche pour segmenter les tumeurs cérébrales même avec la structure compliquée du cerveau, la faible résolution dans certains cas pour les IRM et différents facteurs. Le travail dans les images 2D par l'étape de prétraitement, plus précisément la phase de filtrage, donne une meilleure évaluation, ce qui conduit à un recadrage des IRM de manière bien définie et une meilleure segmentation.

7.4 Perspectives

De nombreuses idées surgissent après chaque recherche, mais toutes nécessitent plus de temps, plus de ressources et d'autres critères. Dans le domaine 3D, des cas d'application de descripteurs multi-échelles et de leur efficacité sont proposés et considérés comme l'un des principaux axes pour continuer à travailler dessus pendant un certain temps.

En même temps, avec l'approche réussie du chapitre 5, son application à différentes applications du monde réel pourrait être suggérée comme mémoire de maîtrise. Pour la partie liée à l'imagerie médicale et aux perspectives d'avenir, nous proposons

de faire de cette segmentation une première étape qui vient avant de classer le grade de la tumeur en étudiant les caractéristiques de la région et en appliquant plusieurs méthodologies récentes dans ce domaine liées aux plusieurs méthodes méta-heuristiques, la méthode SVM (Support Vector Machine), la régression linéaire, et de nombreux autres algorithmes de classification prometteurs.

En plus de le déployer en tant qu'application de bureau pour aider les médecins et les médecins dans les hôpitaux, faciliter la segmentation et afficher les paramètres pertinents nécessaires concernant la zone de la tumeur, le volume, le grade et d'autres spécifications sont trouvés.

Dans cette recherche menée, nous avons essayé de tirer parti de l'outil puissant "représentation graphique" dans deux domaines : les données tridimensionnelles et bidimensionnelles. Les perspectives futures mentionnées sont des suggestions sur lesquelles nous travaillons toujours malgré la fin de la période de doctorat.

Le travail effectué dans cette thèse a donné lieu à une publication dans le journal international de la vision computationnelle et de la robotique et peut être trouvé dans [SCA+22].

En plus d'une présentation orale dans une conférence internationale avec un travail publié présenté dans [SEA+].

Selon le dernier travail lié à la segmentation des tumeurs cérébrales trouvé dans le chapitre 6 nous l'avons soumis au Computer Vision and Image Understanding Journal.

Bibliography

[Abd13] Mostafa Abdelrahman. Geometric modeling of non-rigid 3d shapes: theory and application to object recognition. 2013.

[AE18] Hussna Elnoor Mohammed Abdalla and MY Esmail. Brain tumor detection by using artificial neural network. In *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, pages 1–6. IEEE, 2018.

[AKKS99] Mihael Ankerst, Gabi Kastenmüller, Hans-Peter Kriegel, and Thomas Seidl. 3d shape histograms for similarity search and classification in spatial databases. In *International symposium on spatial databases*, pages 207–226. Springer, 1999.

[AMO93] Ravindra K Ahuja, Thomas L Magnanti, and James B Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice hall, 1993.

[AP19] M Angulakshmi and GG Lakshmi Priya. Walsh hadamard transform for simple linear iterative clustering (slic) superpixel based spectral clustering of multimodal mri brain tumor segmentation. *Irbm*, 40(5):253–262, 2019.

[AS98] Bela BOLLOB AS. Modern graph theory. volume 184 of graduate texts in mathematics. *Springer, New York NY*, 2:4, 1998.

[ASYF20] Javeria Amin, Muhammad Sharif, Mussarat Yasmin, and Steven Lawrence Fernandes. A distinctive approach in brain tumor detection and classification using mri. *Pattern Recognit. Lett.*, 139:118–127, 2020.

[AVB+11] Aitor Aldoma, Markus Vincze, Nico Blodow, David Gossow, Suat Gedikli, Radu Bogdan Rusu, and Gary Bradski. Cad-model recognition and 6dof pose estimation using 3d cues. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pages 585–592. IEEE, 2011.

[BA16] Neslihan Bayramoglu and A Aydın Alatan. Comparison of 3d local and global descriptors for similarity retrieval of range data. *Neurocomputing*, 184:13–27, 2016.

[BAN17] Imran Sarwar Bajwa, Mamoona N Asghar, and M Asif Naeem. Learning-based improved seeded region growing algorithm for brain tumor identification: Improved seeded region growing algorithm for brain tumor identification. *Proceedings of the Pakistan Academy of Sciences: A. Physical and Computational Sciences*, 54(2):127–133, 2017.

[BBO+07] Jan C Buckner, Paul D Brown, Brian P O'Neill, Fredric B Meyer, Cynthia J Wetmore, and Joon H Uhm. Central nervous system tumors. In *Mayo Clinic Proceedings*, volume 82, pages 1271–1286. Elsevier, 2007.

[BBO12] Alexander M Bronstein, Michael M Bronstein, and Maks Ovsjanikov. Feature-based methods in 3d shape analysis. In *3D Imaging, Analysis and Applications*, pages 185–219. Springer, 2012.

[BdLLC20] Thomas Bonald, Nathan de Lara, Quentin Lutz, and Bertrand Charpentier. Scikit-network: Graph analysis in python. *J. Mach. Learn. Res.*, 21(185):1–6, 2020.

[Ben02] E. Bengoetxea. *Inexact Graph Matching Using Estimation of Distribution Algorithms*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, Paris, France, Dec 2002.

[B.L19] T.Ramashri B.Lalitha. Brain mri image segmentation using grab cut algorithm. 2019.

[BM08] John Adrian Bondy and Uppaluri Siva Ramachandra Murty. *Graph theory*. Springer Publishing Company, Incorporated, 2008.

[BMM21] Rouholla Bagheri, Jalal Haghighat Monfared, and Mohammad Reza Montazeriyoun. Brain tumor segmentation using graph coloring approach in magnetic resonance images. *Journal of Medical Signals and Sensors*, 11(4):285, 2021.

[BP12] Dr Samir Kumar Bandhyopadhyay and Tuhin Utsab Paul. Segmentation of brain mri image–a review. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(3), 2012.

[BPK16] Anders G Buch, Henrik G Petersen, and Norbert Krüger. Local shape feature fusion for improved matching, pose estimation and 3d object recognition. *SpringerPlus*, 5(1):297, 2016.

[Bra19] PET Brain. Pregnancy smartsite tm. *Arch Womens Ment Health*, 2019.

[BRT17] Nilesh Bhaskarrao Bahadure, Arun Kumar Ray, and Har Pal Thethi. Image analysis for mri based brain tumor detection and feature extraction using biologically inspired bwt and svm. *International journal of biomedical imaging*, 2017, 2017.

[BS12] Benjamin Bustos and Ivan Sipiran. 3d shape matching for retrieval and recognition. In *3D Imaging, Analysis and Applications*, pages 265–308. Springer, 2012.

[BSAD21] Erena Siyoum Biratu, Friedhelm Schwenker, Yehualashet Megersa Ayano, and Taye Girma Debelee. A survey of brain tumor segmentation and classification algorithms. *Journal of Imaging*, 7(9):179, 2021.

[Bun00] Horst Bunke. Graph matching: Theoretical foundations, algorithms, and applications. In *Proc. Vision Interface*, volume 2000, pages 82–88, 2000.

[BVZ98] Yuri Boykov, Olga Veksler, and Ramin Zabih. Markov random fields with efficient approximations. In *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No. 98CB36231)*, pages 648–655. IEEE, 1998.

[BVZ01] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on pattern analysis and machine intelligence*, 23(11):1222–1239, 2001.

[BZA+18] Mohcine Bouksim, F Rafii Zakani, K Arhid, M Aboulfatah, and T Gadi. New approach for 3d mesh retrieval using data envelopment analysis. *International Journal of Intelligent Engineering and Systems*, 11(1):98–107, 2018.

[CBS05] Timothee Cour, Florence Benezit, and Jianbo Shi. Spectral segmentation with multiscale graph decomposition. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 1124–1131. IEEE, 2005.

[CG12] K Santle Camilus and VK Govindan. A review on graph based segmentation. *International Journal of Image, Graphics and Signal Processing*, 4(5):1, 2012.

[CGF09] Xiaobai Chen, Aleksey Golovinskiy, and Thomas Funkhouser. A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), August 2009.

[Cha19] Navoneel Chakrabarty. Brain mri images for brain tumor detection, Apr 2019.

[CHC+15] Jun Cheng, Wei Huang, Shuangliang Cao, Ru Yang, Wei Yang, Zhaoqiang Yun, Zhijian Wang, and Qianjin Feng. Enhanced performance of brain tumor classification via tumor region augmentation and partition. *PloS one*, 10(10):e0140381, 2015.

[Che17] Jun Cheng. Brain tumor dataset, Apr 2017.

[Cla15] Alex Clark. Pillow (pil fork) documentation, 2015.

[Col13] Andrew Collette. *Python and HDF5*. O'Reilly, 2013.

[CP18] Xinjian Chen and Lingjiao Pan. A survey of graph cuts/graph search based medical image segmentation. *IEEE reviews in biomedical engineering*, 11:112–124, 2018.

[CR09] Victor Chen and Su Ruan. Graph cut based segmentation of brain tumor from mri images. *International Journal on Sciences and Techniques of Automatic control & computer engineering*, 3(2):1054–1063, 2009.

[CR10] Victor Chen and Su Ruan. Graph cut segmentation technique for mri brain tumor extraction. In *2010 2nd International Conference on Image Processing Theory, Tools and Applications*, pages 284–287, 2010.

[Cro16] David F Crouse. On implementing 2d rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696, 2016.

[CV01] Tony F Chan and Luminita A Vese. Active contours without edges. *IEEE Transactions on image processing*, 10(2):266–277, 2001.

[CYH+16] Jun Cheng, Wei Yang, Meiyan Huang, Wei Huang, Jun Jiang, Yujia Zhou, Ru Yang, Jie Zhao, Yanqiu Feng, Qianjin Feng, et al. Retrieval of brain tumors by adaptive spatial pooling and fisher vector representation. *PloS one*, 11(6):e0157112, 2016.

[DCG12] Helin Dutagaci, Chun Pan Cheung, and Afzal Godil. Evaluation of 3d interest point detection techniques via human-generated ground truth. *The Visual Computer*, 28(9):901–917, 2012.

[ECESAB20] Abdallah El Chakik, Abdul Rahman El Sayed, Hassan Alabboud, and Amer Bakkach. An invariant descriptor map for 3d objects matching. *International Journal of Engineering & Technology*, 9(1):59–68, 2020.

[EK72] Jack Edmonds and Richard M Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, 19(2):248–264, 1972.

[FF56] Lester Randolph Ford and Delbert R Fulkerson. Maximal flow through a network. *Canadian journal of Mathematics*, 8:399–404, 1956.

[FH04] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004.

[Gal13] Jean Gallier. Notes on elementary spectral graph theory. applications to graph clustering using normalized cuts. *arXiv preprint arXiv:1311.2492*, 2013.

[GBS+16] Yulan Guo, Mohammed Bennamoun, Ferdous Sohel, Min Lu, Jianwei Wan, and Ngai Ming Kwok. A comprehensive performance evaluation of 3d local feature descriptors. *International Journal of Computer Vision*, 116(1):66–89, 2016.

[GF08] Aleksey Golovinskiy and Thomas Funkhouser. Randomized cuts for 3D mesh analysis. *ACM Transactions on Graphics (Proc. SIGGRAPH ASIA)*, 27(5), December 2008.

[GG86] Stuart Geman and Christine Graffigne. Markov random field image models and their applications to computer vision. In *Proceedings of the international congress of mathematicians*, volume 1, page 2. Berkeley, CA, 1986.

[GGJY20] Chenjie Ge, Irene Yu-Hua Gu, Asgeir Store Jakola, and Jie Yang. Deep semi-supervised learning for brain tumor classification. *BMC Medical Imaging*, 20(1):1–11, 2020.

[GMS13] Nelly Gordillo, Eduard Montseny, and Pilar Sobrevilla. State of the art survey on mri brain tumor segmentation. *Magnetic resonance imaging*, 31(8):1426–1438, 2013.

[GPS89] Dorothy M Greig, Bruce T Porteous, and Allan H Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society: Series B (Methodological)*, 51(2):271–279, 1989.

[GSO00] Gilson A Giraldi, Edilberto Strauss, and Antonio AF Oliveira. A boundary extraction approach based on multi-resolution methods and the t-snakes framework. In *Proceedings 13th Brazilian Symposium on Computer Graphics and Image Processing (Cat. No. PR00878)*, pages 82–89. IEEE, 2000.

[GZWY20] Abubakar Sulaiman Gezawa, Yan Zhang, Qicong Wang, and Lei Yunqi. A review on deep learning approaches for 3d data representations in retrieval and classifications. *IEEE access*, 8:57566–57593, 2020.

[HAM17] Saddam Hussain, Syed Muhammad Anwar, and Muhammad Majid. Brain tumor segmentation using cascaded deep convolutional neural network. In *2017 39th annual International Conference of the IEEE engineering in medicine and biology Society (EMBC)*, pages 1998–2001. IEEE, 2017.

[HHG+20] Rui Hua, Quan Huo, Yaozong Gao, He Sui, Bing Zhang, Yu Sun, Zhanhao Mo, and Feng Shi. Segmenting brain tumor using cascaded v-nets in multimodal mr images. *Frontiers in Computational Neuroscience*, 14:9, 2020.

[HM19] Messaoud Hameurlaine and Abdelouahab Moussaoui. Survey of brain tumor segmentation techniques on magnetic resonance imaging. *Nano Biomed. Eng*, 11(2):178–191, 2019.

[HPPL+12] Paul Heider, Alain Pierre-Pierre, Ruosi Li, Rolf Mueller, and Cindy Grimm. Comparing local shape descriptors. *The Visual Computer*, 28(9):919–929, 2012.

[HPPLG11] Paul Heider, Alain Pierre-Pierre, Ruosi Li, and Cindy Grimm. Local shape descriptors, a survey and evaluation. In *Proceedings of the 4th Eurographics conference on 3D Object Retrieval*, pages 49–56, 2011.

[HSKK01] Masaki Hilaga, Yoshihisa Shinagawa, Taku Kohmura, and Tosiyasu L Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 203–212, 2001.

[IG98] Hiroshi Ishikawa and Davi Geiger. Segmentation by grouping junctions. In *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No. 98CB36231)*, pages 125–131. IEEE, 1998.

[JH98] Andrew E Johnson and Martial Hebert. Surface matching for object recognition in complex three-dimensional scenes. *Image and Vision Computing*, 16(9-10):635–651, 1998.

[KBA+07] PDNL Kleihues, PC Burger, K Aldape, DJ Brat, W Biernat, DD Bigner, Y Nakazato, KH Plate, F Ginagaspero, A von Deimling, et al. Who classification of tumors of the central nervous system. *Lyon: IARC*, pages 33–49, 2007.

[KEBS15] Hany Kasban, MAM El-Bendary, and DH Salama. A comparative study of medical imaging techniques. *International Journal of Information Science and Intelligent System*, 4(2):37–58, 2015.

[Koe90] Jan J Koenderink. *Solid shape*, volume 2. MIT press Cambridge, MA, 1990.

[KPNK03] Marcel Körtgen, Gil-Joo Park, Marcin Novotni, and Reinhard Klein. 3d shape matching with 3d shape contexts. In *The 7th central European seminar on computer graphics*, volume 3, pages 5–17. Citeseer, 2003.

[KS18] Nitu Kumari and Sanjay Saxena. Review of brain tumor segmentation and classification. In *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, pages 1–6. IEEE, 2018.

[Kuh55] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[KVD92] Jan J Koenderink and Andrea J Van Doorn. Surface shape and curvature scales. *Image and vision computing*, 10(8):557–564, 1992.

[KXX19] Zhang Kun, Ma Xiao, and Li Xinguo. Shape matching based on multiscale invariant features. *IEEE Access*, 7:115637–115649, 2019.

[LGB+13] Zhouhui Lian, Afzal Godil, Benjamin Bustos, Mohamed Daoudi, Jeroen Hermans, Shun Kawamura, Yukinori Kurita, Guillaume Lavoué, Hien Van Nguyen, Ryutarou Ohbuchi, et al. A comparison of methods for non-rigid 3d shape retrieval. *Pattern Recognition*, 46(1):449–461, 2013.

[LNJ+17] Graciela Lara López, Adriana Peña Pérez Negrón, Angélica De Antonio Jiménez, Jaime Ramírez Rodríguez, and Ricardo Imbert Paredes. Comparative analysis of shape descriptors for 3d objects. *Multimedia Tools and Applications*, 76(5):6993–7040, 2017.

[MA18] Wan Rosanisah Wan Mohd and Lazim Abdullah. Similarity measures of pythagorean fuzzy sets based on combination of cosine similarity measure and euclidean distance measure. In *AIP conference proceedings*, volume 1974, page 030017. AIP Publishing LLC, 2018.

[MDCA19] Pawel Mlynarski, Hervé Delingette, Antonio Criminisi, and Nicholas Ayache. Deep learning with mixed supervision for brain tumor segmentation. *Journal of Medical Imaging*, 6(3):034002, 2019.

[MEJ20] Mark Willy L Mondia, Adrian I Espiritu, and Roland Dominic G Jamora. Primary brain tumor research productivity in southeast asia and its association with socioeconomic determinants and burden of disease. *Frontiers in Oncology*, 10:607777, 2020.

[MHH+22] Simeon Mayala, Ida Herdlevær, Jonas Bull Haugsøen, Shamundeeswari Anandan, Sonia Gavasso, and Morten Brun. Brain tumor segmentation based on minimum spanning tree. *Frontiers in Signal Processing*, 2, 2022.

[MJD+22] Marco Musy, Guillaume Jacquenot, Giovanni Dalmasso, neoglez, Ruben de Bruin, Ahinoam Pollack, Federico Claudi, Codacy Badger, icemtel, Zhi-Qiang Zhou, Bane Sullivan, Brian Lerner, Daniel Hrisca, Diego Volpatto, Evan, Nico Schlömer, RichardScottOZ, and ilorevilo. marcomusy/vedo: 2022.0.1, January 2022.

[MJF12] Anam Mustaqeem, Ali Javed, and Tehseen Fatima. An efficient brain tumor detection algorithm using watershed & thresholding based segmentation. *International Journal of Image, Graphics and Signal Processing*, 4(10):34, 2012.

[MP07] Pierre Moreels and Pietro Perona. Evaluation of features detectors and descriptors based on 3d objects. *International journal of computer vision*, 73(3):263–284, 2007.

[MPF+18] Fazel Mirzaei, Mohammad Reza Parishan, Mohammadjavad Faridafshin, Reza Faghihi, and Sedigheh Sina. Automated brain tumor segmentation in mr images using a hidden markov classifier framework trained by svd-derived features. *Image Video Process*, 9:1844–1848, 2018.

[OK14] Swe Zin Oo and Aung Soe Khaing. Brain tumor detection and segmentation using watershed segmentation and morphological operation. *International Journal of Research in Engineering and Technology*, 3(03):367–374, 2014.

[Ots79] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[PAH+17] VB Prasath, Haneen Arafat Abu Alfeilat, Ahmad Hassanat, Omar Lasassmeh, Ahmad S Tarawneh, Mahmoud Bashir Alhasanat, and Hamzeh S Eyal Salman. Distance and similarity measures effect on the performance of k-nearest neighbor classifier–a review. *arXiv preprint arXiv:1708.04321*, 2017.

[Pap14] Panagiotis Papadakis. The canonically posed 3d objects dataset. 2014.

[PDCP12] Sarah Parisot, Hugues Duffau, Stéphane Chemouny, and Nikos Paragios. Graph-based detection, segmentation & characterization of brain tumors. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 988–995. IEEE, 2012.

[PJP14] AJ Patil, Prerana Jain, and Ashwini Pachpande. Automatic brain tumor detection using k-means and rflicm. *Int. J. Adv. Res. Electr. Electron. Instrum. Eng*, 3(12):13896–13903, 2014.

[PPAS16] Sérgio Pereira, Adriano Pinto, Victor Alves, and Carlos A Silva. Brain tumor segmentation using convolutional neural networks in mri images. *IEEE transactions on medical imaging*, 35(5):1240–1251, 2016.

[PR99] Eric Paquet and Marc Rioux. Nefertiti: A tool for 3-d shape databases management. *SAE transactions*, pages 387–393, 1999.

[PRM+00] Eric Paquet, Marc Rioux, Anil Murching, Thumpudi Naveen, and Ali Tabatabai. Description of shape information for 2-d and 3-d objects. *Signal processing: Image communication*, 16(1-2):103–122, 2000.

[RAA+19] Muhammad Rafiq, Shahzaib Ashraf, Saleem Abdullah, Tahir Mahmood, and Shakoor Muhammad. The cosine similarity measures of spherical fuzzy sets and their applications in decision making. *Journal of Intelligent & Fuzzy Systems*, 36(6):6059–6073, 2019.

[RBMB08] Radu Bogdan Rusu, Nico Blodow, Zoltan Csaba Marton, and Michael Beetz. Aligning point cloud views using persistent feature histograms. In *2008 IEEE/RSJ international conference on intelligent robots and systems*, pages 3384–3391. IEEE, 2008.

[RBRY19] Reihaneh Rostami, Fereshteh S Bashiri, Behrouz Rostami, and Zeyun Yu. A survey on data-driven 3d shape descriptors. In *Computer Graphics Forum*, volume 38, pages 356–393. Wiley Online Library, 2019.

[RG00] Sebastien Roy and Venu Govindu. Mrf solutions for probabilistic optical flow formulations. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 3, pages 1041–1047. IEEE, 2000.

[S+15] Amritpal Singh et al. Detection of brain tumor in mri images, using combination of fuzzy c-means and svm. In *2015 2nd international conference on signal processing and integrated networks (SPIN)*, pages 98–102. IEEE, 2015.

[SAK+18] Rachida Saouli, Mohamed Akil, Rostom Kachouri, et al. Fully automatic brain tumor segmentation using end-to-end incremental deep neural networks in mri images. *Computer methods and programs in biomedicine*, 166:39–49, 2018.

[SANI16] Md Sujan, Nashid Alam, Syed Abdullah Noman, and Mohammed Jahirul Islam. A segmentation based automated system for brain tumor detection. *International Journal of Computer Applications*, 153(10):41–49, 2016.

[SAO17] Daniel Oliva Sales, Jean Amaro, and Fernando Santos Osório. 3d shape descriptor for objects recognition. In *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, pages 1–6. IEEE, 2017.

[SB11] Ivan Sipiran and Benjamin Bustos. Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *The Visual Computer*, 27(11):963, 2011.

[SCA+22] Roaa Soloh, Abdallah El Chakik, Hassan Alabboud, Ahmad Shahin, and Adnan Yassine. New descriptors' combination for 3d mesh correspondence and retrieval. *International Journal of Computational Vision and Robotics*, 12(4):377–396, 2022.

[SEA+] Roaa SOLOH, Abdallah ELCHAKIK, Hassan ALABBOUD, Ahmad SHAHIN, and Adnan YASSINE. 3d mesh matching using surface descriptor and integer linear programming.

[Sip] Ivan Sipiran. Harris 3d - detecting interest points on 3d meshes.

[SLW+00] Mie Sato, Sarang Lakare, Ming Wan, Arie Kaufman, and Masayuki Nakajima. A gradient magnitude based region growing algorithm for accurate segmentation. In *Proceedings 2000 International Conference on Image Processing (Cat. No. 00CH37101)*, volume 3, pages 448–451. IEEE, 2000.

[SM00] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.

[SMKF04] Philip Shilane, Patrick Min, Michael Kazhdan, and Thomas Funkhouser. The princeton shape benchmark. In *Proceedings Shape Modeling Applications, 2004.*, pages 167–178. IEEE, 2004.

[SP04] Robert W Sumner and Jovan Popović. Deformation transfer for triangle meshes. *ACM Transactions on graphics (TOG)*, 23(3):399–405, 2004.

[SSC20] Paula Stancelova, Elena Sikudova, and Zuzana Cernekova. Performance evaluation of selected 3d keypoint detector–descriptor combinations. In *International Conference on Computer Vision and Graphics*, pages 188–200. Springer, 2020.

[SSGD03] Hari Sundar, Deborah Silver, Nikhil Gagvani, and Sven Dickinson. Skeleton based shape matching and retrieval. In *2003 Shape Modeling International.*, pages 130–139. IEEE, 2003.

[SVZ00] Dan Snow, Paul Viola, and Ramin Zabih. Exact voxel occupancy with graph cuts. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 1, pages 345–352. IEEE, 2000.

[SZM+08] Kaleem Siddiqi, Juan Zhang, Diego Macrini, Ali Shokoufandeh, Sylvain Bouix, and Sven Dickinson. Retrieving articulated 3-d models using medial surfaces. *Machine vision and applications*, 19(4):261–275, 2008.

[Tab11] Hedi Tabia. *Contributions to 3D-shape matching, retrieval and classification.* PhD thesis, 2011.

[TBRN00] Bertrand Thirion, Benedicte Bascle, Visvanathan Ramesh, and Nassir Navab. Fusion of color, shading and boundary information for factory

pipe segmentation. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 349–356. IEEE, 2000.

[TG12] Sarah Tang and Afzal Godil. An evaluation of local shape descriptors for 3d shape retrieval. In *Three-Dimensional Image Processing (3DIP) and Applications II*, volume 8290, page 82900N. International Society for Optics and Photonics, 2012.

[TKR08] Lorenzo Torresani, Vladimir Kolmogorov, and Carsten Rother. Feature correspondence via graph matching: Models and global optimization. In *European conference on computer vision*, pages 596–609. Springer, 2008.

[TLFF17] Zhiqiang Tao, Hongfu Liu, Huazhu Fu, and Yun Fu. Image cosegmentation via saliency-guided constrained clustering with cosine similarity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[TOC10] S. Taheri, S. H. Ong, and V. F. H. Chong. Level-set segmentation of brain tumors using a threshold-based speed function. *Image Vision Comput.*, 28(1):26–37, jan 2010.

[TS02] Orlando José Tobias and Rui Seara. Image segmentation by histogram thresholding using fuzzy sets. *IEEE transactions on Image Processing*, 11(12):1457–1465, 2002.

[TS19] Omid Tarkhaneh and Haifeng Shen. An adaptive differential evolution algorithm to optimal multi-level thresholding for mri brain image segmentation. *Expert Systems with Applications*, 138:112820, 2019.

[TV08] Johan WH Tangelder and Remco C Veltkamp. A survey of content based 3d shape retrieval methods. *Multimedia tools and applications*, 39(3):441–471, 2008.

[uAJC14] Qurat ul Ain, M. Arfan Jaffar, and Tae-Sun Choi. Fuzzy anisotropic diffusion based segmentation and texture based ensemble classification of brain tumor. *Appl. Soft Comput.*, 21:330–340, 2014.

[VA15] KB Vaishnavee and K Amshakala. An automated mri brain image segmentation and tumor detection using som-clustering and proximal support vector machine classifier. In *2015 IEEE international conference on engineering and technology (ICETECH)*, pages 1–6. IEEE, 2015.

[Vek00] Olga Veksler. Image segmentation by nested cuts. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 1, pages 339–344. IEEE, 2000.

[VGO+20] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.

[VKZHCO11] Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. In *Computer Graphics Forum*, volume 30, pages 1681–1707. Wiley Online Library, 2011.

[VL07] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[Vol89] A Volgenant. Teaching linear assignment by mack's algorithm. *Twenty-five Years of Operations Research in the Netherlands: Papers Dedicated to Gijs de Leve*, 70:54, 1989.

[VRDJ95] Guido Van Rossum and Fred L Drake Jr. *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.

[WBV19] Anjali Wadhwa, Anuj Bhardwaj, and Vivek Singh Verma. A review on brain tumor segmentation of mri images. *Magnetic resonance imaging*, 61:247–259, 2019.

[Wik22] Wikipedia. Image segmentation — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Image%20segmentation&oldid=1085113705, 2022. [Online; accessed 12-June-2022].

[WNS+13] Lisa Woodbine, Jessica A. Neal, Nanda-Kumar Sasi, Mayuko Shimada, Karen Deem, Helen Coleman, William B. Dobyns, Tomoo Ogi, Katheryn Meek, E. Graham Davies, and Penny A. Jeggo. Prkdc mutations in a scid patient with profound neurological abnormalities. *The Journal of Clinical Investigation*, 123(7):2969–2980, 7 2013.

[WOBL17] Song Wu, Ard Oerlemans, Erwin M Bakker, and Michael S Lew. A comprehensive evaluation of local detectors and descriptors. *Signal Processing: Image Communication*, 59:150–167, 2017.

[Won05] Koon-Pong Wong. Medical image segmentation: methods and applications in functional imaging. In *Handbook of biomedical image analysis*, pages 111–182. Springer, 2005.

[WV11] Walter Wohlkinger and Markus Vincze. Shape distributions on voxel surfaces for 3d object classification from depth images. In *2011 IEEE international conference on signal and image processing applications (ICSIPA)*, pages 115–120. IEEE, 2011.

[XP98] Chenyang Xu and Jerry L Prince. Generalized gradient vector flow external forces for active contours. *Signal processing*, 71(2):131–139, 1998.

[YH09] Jae-Chern Yoo and Tae Hee Han. Fast normalized cross-correlation. *Circuits, systems and signal processing*, 28(6):819–843, 2009.

[ZdFFeR07] Lisha Zhang, M João da Fonseca, Alfredo Ferreira, and Combinando Realidade Aumentada e Recuperaçao. Survey on 3d shape descriptors. *FundaÃgao para a Cincia ea Tecnologia, Lisboa, Portugal, Tech. Rep. Technical Report, DecorAR (FCT POSC/EIA/59938/2004)*, 3, 2007.

[ZL04] Dengsheng Zhang and Guojun Lu. Review of shape representation and description techniques. *Pattern recognition*, 37(1):1–19, 2004.

[ZSS+19] Muhammad Zawish, Asad Ali Siyal, Shahzad Hyder Shahani, Aisha Zahid Junejo, and Aiman Khalil. Brain tumor segmentation through region-based, supervised and unsupervised learning methods: A literature survey. *Journal of biomedical engineering and medical imaging*, 6(2):08–26, 2019.

[ZTC+20] Ralph T Zade, Jason P Tartaglione, Ernest Chisena, Curtis T Adams, and Matthew R DiCaprio. The quality of online orthopaedic oncology information. *JAAOS Global Research & Reviews*, 4(3), 2020.

[ZTZ15] AA Zakharov, A Yu Tuzhilkin, and AL Zhiznyakov. Finding correspondences between images using descriptors and graphs. *Procedia engineering*, 129:391–396, 2015.