УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА

Vladimir Ilić

Application of New Shape Descriptors And Theory of Uncertainty in Image Processing

DOCTORAL DISSERTATION

Владимир Илић

Примена нових дескриптора облика и теорије неодређености у обради слике

ДОКТОРСКА ДИСЕРТАЦИЈА

Нови Сад, 2019.

УНИВЕРЗИТЕТ У НОВОМ САДУ ● **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**
21000 НОВИ САД, Трг Доситеја Обрадовића 6

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, **РБР**:	
Идентификациони број, **ИБР**:	
Тип документације, **ТД**:	Монографска документација
Тип записа, **ТЗ**:	Текстуални штампани материјал
Врста рада, **ВР**:	Докторска дисертација
Аутор, **АУ**:	Владимир Илић
Ментор, **МН**:	Проф. др Небојша Ралевић
Наслов рада, **НР**:	Примена нових дескриптора облика и теорије неодређености у обради слике
Језик публикације, **ЈП**:	Енглески
Језик извода, **ЈИ**:	Српски, Енглески
Земља публиковања, **ЗП**:	Република Србија
Уже географско подручје, **УГП**:	Аутономна Покрајина Војводина
Година, **ГО**:	2019
Издавач, **ИЗ**:	Ауторски репринт
Место и адреса, **МА**:	Нови Сад, Факултет техничких наука, Трг Доситеја Обрадовића 6
Физички опис рада, **ФО**: (поглавља/страна/цитата/табела/слика/графика/прилога)	8/209/185/11/49/12/0
Научна област, **НО**:	Математичке науке
Научна дисциплина, **НД**:	Анализа облика, Теорија неодређености, Обрада слике
Предметна одредница/Кључне речи, **ПО**:	Облик, Дескриптори облика, Инваријанте облика, Инваријанте фази облика, Класификација објеката, Теорија неодређености, Теорија фази скупова, Дигитална фази репрезентација слике, Модел покривености, Функције растојања, Трансформације растојањима
УДК	
Чува се, **ЧУ**:	Библиотека Факултета техничких наука, Трг Доситеја Обрадовића 6, Нови Сад
Важна напомена, **ВН**:	

Извод, **ИЗ**:		Докторска дисертација се бави проучавањем квантитативних аспеката атрибута облика погодних за нумеричку карактеризацију, то јест дескриптора облика, као и теоријом неодређености, посебно теоријом фази скупова, и њиховом применом у обради слике. Оригинални доприноси и резултати тезе могу се природно поделити у две групе, у складу са приступом и методологијом која је коришћена за њихово добијање. Прва група доприноса односи се на увођење нових дескриптора облика (шестоугаоности и фази квадратности) као и одговарајућих мера које нумерички оцењују у ком обиму разматрани облик задовољава разматрана својства. Уведене мере су природно дефинисане, теоријски добро засноване и задовољавају већину пожељних својстава које свака добро дефинисана мера облика треба да задовољава. Поменимо неке од њих: обе мере узимају вредности из интервала (0,1] и достижу највећу могућу вредност 1 ако и само ако је облик који се посматра шестоугао, односно фази квадрат; не постоји облик не-нула површине чија је измерена шестоугаоност, односно фази квадратност једнака 0; обе уведене мере су инваријантне у односу на трансформације сличности; и дају резултате који су у складу са теоријски доказаним резултатима, као и људском перцепцијом и очекивањима. Бројни експерименти на синтетичким и реалним примерима приказани су у циљу илустровања теоријски доказаних разматрања и пружања јаснијег увида у понашање уведених мера. Њихова предност и корисност илустровани су у различитим задацима препознавања и класификације слика објеката неколико познатих и најчешће коришћених база слика. Поред тога, докторска теза садржи истраживања везана за примену теорије неодређености, у ужем смислу теорије фази скупова, у различитим задацима обраде слике и анализе облика. Разликујемо задатке који се односе на издвајање карактеристика облика и оне који се односе на побољшање перформанси различитих техника обраде и анализе слике. Што се тиче прве групе задатака, бавимо се применом теорије фази скупова у задацима дефинисања новог дескриптора фази облика, назван фази квадратност, и мерења колико је фази квадратан посматрани фази облик. У другој групи задатака бавимо се истраживањем побољшања перформанси оцене трансформације слике еуклидским растојањима у три димензије (3Д ЕДТ), као и сигнатуре непрекидног облика у две димензије засноване на растојању од центроида облика. Ово последње се посебно огледа у постигнутој тачности и прецизности оцене, повећаној инваријантности у односу на ротацију и транслацију објекта, као и робустности у присуству шума и неодређености које су последица несавршености уређаја или услова снимања. Последњи резултати се такође односе и на другу групу оригиналних доприноса тезе који су мотивисани чињеницом да анализа облика традиционално претпоставља да су објекти на слици претходно једнозначно и јасно издвојени из слике. Такво издвајање објеката се обично постиже у процесу јасне (то јест бинарне) сегментације оригиналне слике где се одлука о припадности тачке објекту на слици доноси на једнозначан и недвосмислени начин. Међутим, услед несавршености услова или уређаја за снимање, присуства шума и различитих врста непрецизности (на пример непостојање прецизне границе објекта или јасних граница између самих објеката, грешке у рачунању, недостатка информација, итд.), могу се појавити различити нивои несигурности и неодређености у процесу доношења одлуке у вези са припадношћу тачке слике. Ово је посебно видљиво у случају дискретизације (то јест узорковања) непрекидног домена слике када елемент слике, придружен одговарајућој тачки узорка домена, може бити делимично покривен са више објеката на слици. У том смислу, имамо да ова врста сегментације може потенцијално довести до погрешне одлуке о припадности тачака слике, а самим тим и неповратног губитка информација о објектима који се на слици налазе. То произлази из чињенице да сегментација слике изведена на овај начин не дозвољава да тачка слике може делимично у одређеном обиму бити члан посматраног објекта на слици, што даље води потенцијалном ризику да тачке делимично садржане у објекту пре сегментације неће бити придружене објекту након сегментације. Међутим, ако се уместо бинарне сегментације изврши сегментација слике где се одлука о припадности тачке слике објекту доноси на начин који омогућава да тачка може делимично бити члан објекта у неком обиму, тада се доношење бинарне одлуке о чланство тачке објекту на слици може избећи у овом раном кораку анализе. То даље резултира да се потенцијално велика количина информација о објектима присутним на слици може сачувати након сегментације, и користити у следећим корацима анализе. С тим у вези, од посебног интереса за нас јесте специјална врста фази сегментације слике, сегментација заснована на покривености елемената слике, која као резултат обезбеђује фази дигиталну репрезентацију слике где је вредност чланства додељена сваком елементу пропорционална његовој релативној покривености непрекидним објектом на оригиналној слици. У овој тези бавимо се истраживањем модела дигитализације покривености који пружа овакву врсту репрезентацију слике и представљамо како се могу постићи значајна побољшања у оцени 3Д ЕДТ, као и сигнатуре непрекидног облика засноване на растојању од центроида, ако су информације о покривености доступне у овој репрезентацији слике разматране на одговарајући начин.	
Датум прихватања теме, **ДП**:		28.03.2019.	
Датум одбране, **ДО**:			
Чланови комисије, **КО**:	Председник:	др Силвиа Гилезан, редовни професор	
	Члан:	др Бранислав Боровац, редовни професор	
	Члан:	др Мирјана Ђорић, редовни професор	
	Члан:	др Тибор Лукић, ванредни професор	Потпис ментора
	Члан:	др Лидија Чомић, доцент	
	Члан, ментор:	др Небојша Ралевић, редовни професор	

UNIVERSITY OF NOVI SAD ● **FACULTY OF TECHNICAL SCIENCES**
21000 NOVI SAD, Trg Dositeja Obradovića 6

KEY WORDS DOCUMENTATION

Accession number, **ANO**:	
Identification number, **INO**:	
Document type, **DT**:	Monographic publication
Type of record, **TR**:	Textual printed material
Contents code, **CC**:	PhD thesis
Author, **AU**:	Vladimir Ilić
Mentor, **MN**:	Professor Nebojša Ralević, PhD
Title, **TI**:	Application of new shape descriptors and theory of uncertainty in image processing
Language of text, **LT**:	English
Language of abstract, **LA**:	Serbian, English
Country of publication, **CP**:	Republic of Serbia
Locality of publication, **LP**:	Autonomous Province of Vojvodina
Publication year, **PY**:	2019
Publisher, **PB**:	Author's reprint
Publication place, **PP**:	Novi Sad, Faculty of Technical Sciences, Trg Dositeja Obradovića 6
Physical description, **PD**: (chapters/pages/ref./tables/pictures/graphs/appendixes)	8/209/185/11/49/12/0
Scientific field, **SF**:	Mathematical sciences
Scientific discipline, **SD**:	Shape analsys, Theory of uncertainty, Image processing
Subject/Key words, **S/KW**:	Shape, Shape descriptors, Shape Invariants, Fuzzy shape invariants, Object classification, Theory of uncertainty, Fuzzy set theory, Fuzzy digital image representation, Coverage model, Distance functions, Distance transforms
UC	
Holding data, **HD**:	Library of the Faculty of Technical Sciences, Trg Dositeja Obradovića 6, Novi Sad
Note, **N**:	

UNIVERSITY OF NOVI SAD ● **FACULTY OF TECHNICAL SCIENCES**
21000 NOVI SAD, Trg Dositeja Obradovića 6

KEY WORDS DOCUMENTATION

Abstract, **AB**:

The doctoral thesis deals with the study of quantitative aspects of shape attributes suitable for numerical characterization, i.e., shape descriptors, as well as the theory of uncertainty, particularly the theory of fuzzy sets, and their application in image processing. The original contributions and results of the thesis can be naturally divided into two groups, in accordance with the approaches used to obtain them. The first group of contributions relates to introducing new shape descriptors (of hexagonality and fuzzy squareness) and associated measures that evaluate to what extent the shape considered satisfies these properties. The introduced measures are naturally defined, theoretically well-founded, and satisfy most of the desirable properties expected to be satisfied by each well-defined shape measure. To mention some of them: they both range through (0,1] and achieve the largest possible value 1 if and only if the shape considered is a hexagon, respectively a fuzzy square; there is no non-zero area shape with the measured hexagonality or fuzzy squareness equal to 0; both introduced measures are invariant to similarity transformations; and provide results that are consistent with the theoretically proven results, as well as human perception and expectation. Numerous experiments on synthetic and real examples are shown aimed to illustrate theoretically proven considerations and to provide clearer insight into the behaviour of the introduced shape measures. Their advantages and applicability are illustrated in various tasks of recognizing and classifying objects images of several well-known and most frequently used image datasets. Besides, the doctoral thesis contains research related to the application of the theory of uncertainty, in the narrower sense fuzzy set theory, in the different tasks of image processing and shape analysis. We distinguish between the tasks relating to the extraction of shape features, and those relating to performance improvement of different image processing and image analysis techniques. Regarding the first group of tasks, we deal with the application of fuzzy set theory in the tasks of introducing new fuzzy shape-based descriptor, named fuzzy squareness, and measuring how much fuzzy square is given fuzzy shape. In the second group of tasks, we deal with the study of improving the performance of estimates of both the Euclidean distance transform in three dimensions (3D EDT) and the centroid distance signature of shape in two dimensions. Performance improvement is particularly reflected in terms of achieved accuracy and precision, increased invariance to geometrical transformations (e.g., rotation and translation), and robustness in the presence of noise and uncertainty resulting from the imperfection of devices or imaging conditions.

The latter also refers to the second group of the original contributions and results of the thesis. It is motivated by the fact that the shape analysis traditionally assumes that the objects appearing in the image are previously uniquely and crisply extracted from the image. This is usually achieved in the process of sharp (i.e., binary) segmentation of the original image where a decision on the membership of point to an imaged object is made in a sharp manner. Nevertheless, due to the imperfections of imaging conditions or devices, the presence of noise, and various types of imprecision (e.g., lack of precise object boundary or clear boundaries between the objects, errors in computation, lack of information, etc.), different levels of uncertainty and vagueness in the process of making a decision regarding the membership of image point may potentially occur. This is particularly noticeable in the case of discretization (i.e., sampling) of continuous image domain when a single image element, related to corresponding image sample point, is covered by multiple objects in an image. In this respect, it is clear that this type of segmentation can potentially lead to a wrong decision on the membership of image points, and consequently irreversible information loss about the imaged objects. This stems from the fact that image segmentation performed in this way does not permit that the image point may be a member to a particular imaged object to some degree, further leading to the potential risk that points partially contained in the object before segmentation will not be assigned to the object after segmentation. However, if instead of binary segmentation, it is performed segmentation where a decision about the membership of image point is made in a gradual rather than crisp manner, enabling that point may be a member to an object to some extent, then making a sharp decision on the membership can be avoided at this early analysis step. This further leads that potentially a large amount of object information can be preserved after segmentation and used in the following analysis steps. In this regard, we are interested in one specific type of fuzzy segmentation, named coverage image segmentation, resulting in fuzzy digital image representation where membership value assigned to each image element is proportional to its relative coverage by a continuous object present in the original image. In this thesis, we deal with the study of coverage digitization model providing coverage digital image representation and present how significant improvements in estimating 3D EDT, as well as the centroid distance signature of continuous shape, can be achieved, if the coverage information available in this type of image representation is appropriately considered.

Accepted by the Scientific Board on, **ASB**:		28.03.2019.	
Defended on, **DE**:			
Defended Board, **DB**:	President:	Silvia Gilezan, PhD, full professor	
	Member:	Branislav Borovac, PhD, full professor	
	Member:	Mirjana Đorić, PhD, full professor	
	Member:	Tibor Lukić, PhD, associate professor	Menthor's sign
	Member:	Lidija Čomić, PhD, assistant professor	
	Member, Mentor:	Nebojša Ralević, PhD, full professor	

Acknowledgements

I dedicate PhD to my family who was with me at all times and gave me selfless support and love. Thank you to wife Jelena, son Andrei and daughter Anastasia for the warmth, love, understanding and unconditional support you have given me daily. Thank you to father Miroslav, mother Živadinka and sister Aleksandra for their understanding and advice for encouraging and supporting me.

Special thanks to my mentor prof. dr Nebojša Ralević. Thank you, professor, for the confidence and support you have given me unreservedly, for the knowledge and experience you have selflessly imparted, the helpful advice and understanding, for the energy and effort you have put into me, my teaching and research work, the time you have spent for me, pleasant companionship, and wholehearted help.

I would like to thank prof. dr Silvia Gilezan, on the understanding, support and positive energy that she encouraged and encouraged me during many years of working together at the Faculty of Technical Sciences. Thanks for the friendly companionship, helpful tips, and support I can count on at any time.

I also owe my thanks to prof. dr Mila Stojaković, the head of Chair of Mathematics at the Faculty of Technical Sciences, for understanding, honest and open conversations, relentless positive energy that she passed on to me every day and motivated me to cope with various challenges for the benefit of myself and others.

I would also like to thank all the members of the committee for the evaluation and defense of the doctoral dissertation for their thorough reading of this dissertation and for the helpful remarks, suggestions, and affirmative advice that greatly contributed to improving the quality of the manuscript of the dissertation itself.

The author

Zahvalnica

Doktorat posvećujem svojoj porodici koja je u svakom trenutku bila uz mene i pružila mi nesebičnu podršku i ljubav. Hvala supruzi Jeleni, sinu Andreju i kćerki Anastasiji na toplini, ljubavi, razumevanju i bezuslovnoj podršci koju ste mi svakodnevno pružili. Hvala ocu Miroslavu, majci Živadinki i sestri Aleksandri na razumevanju i savetima kojima ste me ohrabrivali i podržavali.

Posebnu zahvalnost u izradi doktorske disertacije dugujem svom mentoru prof. dr Nebojši Raleviću. Hvala Vam, profesore, na ukazanom poverenju i podršci koju ste mi bezrezervno pružili, za znanja i iskustva koja ste mi nesebično prenosili, korisnim savetima i razumevanju, za energiju i trud koji ste uložili u mene i moj nastavni i naučno-istraživački rad, na izdvojenom vremenu, prijatnom druženju i svesrdnoj pomoći.

Zahvaljujem se posebno i prof. dr Silvii Gilezan na razumevanju, podršci i pozitivnoj energiji kojom me je ohrabrivala i podsticala tokom dugogodišnjeg zajedničkog rada na Fakultetu tehničkih nauka. Hvala na prijatnom druženju, korisnim savetima, i podršci na koju sam u svakom trenutku mogao da računam.

Zahvalnost dugujem i prof. dr Mili Stojaković, šefu Katedre za matematiku na Fakultetu tehničkih nauka, na razumevanju, iskrenim i otvorenim razgovorima, neumornoj pozitivnoj energiji koju je svakodnevno prenosila na mene i motivisala me kako da sa raznim izazovima izađem na kraj na korist sebi i drugima.

Takođe, želim da se zahvalim i svim članovima komisije za ocenu i odbranu doktorske disertacije na detaljno iščitanom rukopisu ove disertacije i korisnim primedbama, sugestijama i afirmativnim savetima koji su u velikoj meri doprineli poboljšanju kvaliteta same doktorske disertacije.

Autor

Abstract

The doctoral thesis deals with the study of quantitative aspects of shape attributes suitable for numerical characterization, i.e., shape descriptors, as well as the theory of uncertainty, particularly the theory of fuzzy sets, and their application in image processing. The original contributions and results of the thesis can be naturally divided into two groups, in accordance with the approaches used to obtain them. The first group of contributions relates to introducing new shape descriptors (of hexagonality and fuzzy squareness) and associated measures that evaluate to what extent the shape considered satisfies these properties. The introduced measures are naturally defined, theoretically well-founded, and satisfy most of the desirable properties expected to be satisfied by each well-defined shape measure. To mention some of them: they both range through $(0, 1]$ and achieve the largest possible value 1 if and only if the shape considered is a hexagon, respectively a fuzzy square; there is no non-zero area shape with the measured hexagonality or fuzzy squareness equal to 0; both introduced measures are invariant to similarity transformations; and provide results that are consistent with the theoretically proven results, as well as human perception and expectation. Numerous experiments on synthetic and real examples are shown aimed to illustrate theoretically proven considerations and to provide clearer insight into the behaviour of the introduced shape measures. Their advantages and applicability are illustrated in various tasks of recognizing and classifying objects images of several well-known and most frequently used image datasets. Besides, the doctoral thesis contains research related to the application of the theory of uncertainty, in the narrower sense fuzzy set theory, in the different tasks of image processing and shape analysis. We distinguish between the tasks relating to the extraction of shape features, and those relating to performance improvement of different image processing and image analysis techniques. Regarding the first group of tasks, we deal with the application of fuzzy set theory in the tasks of introducing new fuzzy shape-based descriptor, named fuzzy squareness, and measuring how much fuzzy square is given fuzzy shape. In the second group of tasks, we deal with the study of improving the performance of estimates of both the Euclidean distance transform in three dimensions (3D EDT) and the centroid distance signature of shape in two dimensions. Performance improvement is

particularly reflected in terms of achieved accuracy and precision, increased invariance to geometrical transformations (e.g., rotation and translation), and robustness in the presence of noise and uncertainty resulting from the imperfection of devices or imaging conditions.

The latter also refers to the second group of the original contributions and results of the thesis. It is motivated by the fact that the shape analysis traditionally assumes that the objects appearing in the image are previously uniquely and crisply extracted from the image. This is usually achieved in the process of sharp (i.e., binary) segmentation of the original image where a decision on the membership of point to an imaged object is made in a sharp manner. Nevertheless, due to the imperfections of imaging conditions or devices, the presence of noise, and various types of imprecision (e.g., lack of precise object boundary or clear boundaries between the objects, errors in computation, lack of information, etc.), different levels of uncertainty and vagueness in the process of making a decision regarding the membership of image point may potentially occur. This is particularly noticeable in the case of discretization (i.e., sampling) of continuous image domain when a single image element, related to corresponding image sample point, is covered by multiple objects in an image. In this respect, it is clear that this type of segmentation can potentially lead to a wrong decision on the membership of image points, and consequently irreversible information loss about the imaged objects. This stems from the fact that image segmentation performed in this way does not permit that the image point may be a member to a particular imaged object to some degree, further leading to the potential risk that points partially contained in the object before segmentation will not be assigned to the object after segmentation. However, if instead of binary segmentation, it is performed segmentation where a decision about the membership of image point is made in a gradual rather than crisp manner, enabling that point may be a member to an object to some extent, then making a sharp decision on the membership can be avoided at this early analysis step. This further leads that potentially a large amount of object information can be preserved after segmentation and used in the following analysis steps. In this regard, we are interested in one specific type of fuzzy segmentation, named coverage image segmentation, resulting in fuzzy digital image representation where membership value assigned to each image element is proportional to its relative coverage by a continuous object present in the original image. In this thesis, we deal with the study of coverage digitization model providing coverage digital image representation and present how significant improvements in estimating 3D EDT, as well as the centroid distance signature of continuous shape, can be achieved, if the coverage information available in this type of image representation is appropriately considered.

Rezime

Doktorska disertacija se bavi proučavanjem kvantitativnih aspekata atributa oblika pogodnih za numeričku karakterizaciju, to jest deskriptora oblika, kao i teorijom neodređenosti, posebno teorijom fazi skupova, i njihovom primenom u obradi slike. Originalni doprinosi i rezultati teze mogu se prirodno podeliti u dve grupe, u skladu sa pristupom i metodologijom koja je korišćena za njihovo dobijanje. Prva grupa doprinosa odnosi se na uvođenje novih deskriptora oblika (šestougaonosti i fazi kvadratnosti) kao i odgovarajućih mera koje numerički ocenjuju u kom obimu razmatrani oblik zadovoljava razmatrana svojstva. Uvedene mere su prirodno definisane, teorijski dobro zasnovane i zadovoljavaju većinu poželjnih svojstava koje svaka dobro definisana mera oblika treba da zadovoljava. Pomenimo neke od njih: obe mere uzimaju vrednosti iz intervala $(0, 1]$ i dostižu najveću moguću vrednost 1 ako i samo ako je oblik koji se posmatra šestougao, odnosno fazi kvadrat; ne postoji oblik ne-nula površine čija je izmerena šestougaonost, odnosno fazi kvadratnost jednaka 0; obe uvedene mere su invarijantne u odnosu na transformacije sličnosti; i daju rezultate koji su u skladu sa teorijski dokazanim rezultatima, kao i ljudskom percepcijom i očekivanjima. Brojni eksperimenti na sintetičkim i realnim primerima prikazani su u cilju ilustrovanja teorijski dokazanih razmatranja i pružanja jasnijeg uvida u ponašanje uvedenih mera. Njihova prednost i korisnost ilustrovani su u različitim zadacima prepoznavanja i klasifikacije slika objekata nekoliko poznatih i najčešće korišćenih baza slika. Pored toga, doktorska teza sadrži istraživanja vezana za primenu teorije neodređenosti, u užem smislu teorije fazi skupova, u različitim zadacima obrade slike i analize oblika. Razlikujemo zadatke koji se odnose na izdvajanje karakteristika oblika i one koji se odnose na poboljšanje performansi različitih tehnika obrade i analize slike. Što se tiče prve grupe zadataka, bavimo se primenom teorije fazi skupova u zadacima definisanja novog deskriptora fazi oblika, nazvan fazi kvadratnost, i merenja koliko je fazi kvadratan posmatrani fazi oblik. U drugoj grupi zadataka bavimo se istraživanjem poboljšanja performansi ocene transformacije slike euklidskim rastojanjima u tri dimenzije (3D EDT), kao i signature neprekidnog oblika u dve dimenzije zasnovane na rastojanju od centroida oblika. Ovo poslednje se posebno ogleda u postignutoj tačnosti i preciznosti ocene, povećanoj invarijantnosti

u odnosu na rotaciju i translaciju objekta, kao i robustnosti u prisustvu šuma i neodređenosti koje su posledica nesavršenosti uređaja ili uslova snimanja.

Poslednji rezultati se takođe odnose i na drugu grupu originalnih doprinosa teze koji su motivisani činjenicom da analiza oblika tradicionalno pretpostavlja da su objekti na slici prethodno jednoznačno i jasno izdvojeni iz slike. Takvo izdvajanje objekata se obično postiže u procesu jasne (to jest binarne) segmentacije originalne slike gde se odluka o pripadnosti tačke objektu na slici donosi na jednoznačan i nedvosmisleni način. Međutim, usled nesavršenosti uslova ili uređaja za snimanje, prisustva šuma i različitih vrsta nepreciznosti (na primer nepostojanje precizne granice objekta ili jasnih granica između samih objekata, grešaka u računanju, nedostatka informacija, itd.), mogu se pojaviti različiti nivoi nesigurnosti i neodređenosti u procesu donošenja odluke u vezi sa pripadnošću tačke slike. Ovo je posebno vidljivo u slučaju diskretizacije (to jest uzorkovanja) neprekidnog domena slike kada element slike, pridružen odgovarajućoj tački uzorka domena, može biti delimično pokriven sa više objekata na slici. U tom smislu, imamo da ova vrsta segmentacije može potencijalno dovesti do pogrešne odluke o pripadnosti tačaka slike, a samim tim i nepovratnog gubitka informacija o objektima koji se na slici nalaze. To proizlazi iz činjenice da segmentacija slike izvedena na ovaj način ne dozvoljava da tačka slike može delimično u određenom obimu biti član posmatranog objekta na slici, što dalje vodi potencijalnom riziku da tačke delimično sadržane u objektu pre segmentacije neće biti pridružene objektu nakon segmentacije. Međutim, ako se umesto binarne segmentacije izvodi segmentacija gde se odluka o pripadnosti tačke slike objektu donosi na način koji omogućava da tačka može biti član objekta u nekom stepenu, tada se donošenje binarne odluke o članstvo tačke objektu na slici može izbeći u ovom ranom koraku analize. To dalje rezultira da se potencijalno velika količina informacija o objektima prisutnim na slici može sačuvati nakon segmentacije, i koristiti u sledećim koracima analize. S tim u vezi, od posebnog interesa za nas jeste specijalna vrsta fazi segmentacije slike, pokrivenost segmentacija slike, koja kao rezultat obezbeđuje fazi digitalnu reprezentaciju slike gde je vrednost članstva dodeljena svakom elementu slike proporcionalna njegovoj relativnoj pokrivenosti neprekidnim objektom na originalnoj slici. U ovoj tezi bavimo se istraživanjem modela digitalizacije pokrivenosti koji pruža ovakvu vrstu reprezentaciju slike i predstavljamo kako se mogu postići značajna poboljšanja u oceni 3D EDT, kao i signature neprekidnog oblika zasnovane na rastojanju od centroida, ako su informacije o pokrivenosti dostupne u ovoj reprezentaciji slike razmatrane na odgovarajući način.

S obzirom da ljudski vizuelni sistem razume oblik kao jednu od važnih komponenti neophodnih za opažanje i prepoznavanje objekata koji nas okružuju, onda i ne iznenađuje činjenica zašto analiza oblika predstavlja jedan od veoma važnih koraka (nakon dobijanja slike, njene potencijalne predobrade i izdvajanja objekata, odnosno segmentacija slike) u kojoj su određene karakteristike objekta izdvajaju, a zatim koriste u narednim koracima analize slike. To se ne odnosi samo na određene jednos-

tavne zadatke prepoznavanja geometrijskih oblika, napisanih brojeva ili odštampanih karaktera, već i na druge komplikovanije zadatke kompjuterske vizije, gde analiza oblika predstavlja samo jedan od neophodnih koraka pre nego što se primeni određeni algoritam veštačke inteligencije ili rutina mašinskog učenja. To sledi iz činjenice da u različitim oblastima istraživanja i primena, zahvaljujući brzom razvoju tehnologija za snimanje slika, imamo priliku da radimo sa velikom količinom podataka zasnovanih na slici. Izazovi rada sa ovim slikama mogu biti višestruki s obzirom da postoji stalna potreba za njihovom analizom, obradom, upoređivanjem ili klasifikacijom. U svim tim zadacima, u zavisnosti od primena i zahteva postavljenih ispred nas, potrebno je identifikovati i razumeti kontekst objekata na slici kako bi sadržaj same slike bio razumljiv. Shodno tome, kao odgovor na ove izazove, do sada je razvijeno nekoliko različitih pristupa i računarskih metoda sa teorijskog i eksperimentalnog stanovišta. S tim u vezi, trebalo bi napomenuti da značaj i prednosti ovih metoda prvenstveno zavise od njihove primenljivosti u različitim oblastima istraživanja. Pomenimo samo nekoliko: poljoprivreda [95, 102, 145, 163], astronomija [49, 65], biologija [93, 100], botanika [41, 77, 84], ekologija [11], geografija [5, 168], medicina [28, 55, 106, 114], robotika [56, 98], daljinsko očitavanje [132–134, 177], transport [14, 52, 62, 126], itd.

U ovoj tezi posebnu pažnju usmeravamo na karakterizaciju objekta koja se zasniva na analizi oblika, imajući u vidu da oblik kao jedna od elementarnih vizuelnih osobina objekta (zajedno sa teksturom i bojom) ima različite atribute koji se mogu numerički oceniti, i, shodno tome, iskoristiti za karakterizaciju objekta. Oblik se obično se predstavlja kao ograničena, ne nužno povezana, oblast u ravni ili prostoru koja određuje prostorni opseg objekta, odnosno njegov nosač, kada su iz njega isključene njegove vizuelne karakteristike poput teksture i boje. Kao takav, oblik se može identifikovati sa siluetom objekta koja se dobija njegovim osvetljenjem beskonačno dalekim izvorom svetlosti [94]. Važnost oblika kao alata za analizu i karakterizaciju objekta dovela je do njegovog proučavanja u različitim zadacima analize objekata i rezultirala obimnom literaturom koja se odnosi na ovu temu [9, 47, 67, 83, 109, 116, 125, 139, 166]. U nastavku ćemo predstaviti nekoliko različitih alata u analizi oblika koji se razlikuju po osnovu nekoliko kriterijuma, što će nam omogućiti da razmatrane alate i metode, korišćene u tezi, možemo na odgovarajući način klasifikovati. Osnovna podela između različitih pristupa u analizi i karakterizaciji oblika tiče se načina kako se karakteristike oblika izdvajaju iz samog oblika, odnosno koje tačke oblika se koriste u njegovoj karakterizaciji [167, 181]. S tim u vezi razlikujemo tehnike *zasnovane na celoj oblasti oblika* koje koriste sve tačke oblika i one *zasnovane na granici oblika* koje koriste samo tačke granice oblika [94]. U tom kontekstu, tehnike zasnovane na oblasti su globalne po svojoj prirodi i uglavnom su usmerene na globalnu analizu oblika radi njegove karakterizacije. Kao takvi pokazali su se korisnim u radu sa generičkim oblicima i podacima slabijeg kvaliteta, i takođe su i stabilni u prisustvu šuma. Među najčešće korišćenim metodama zasnovanim na oblasti su moment invarijante [66, 153, 172], složeni deskriptor slike [87], generički Furijeovi deskrip-

tori [180], Zernike momenti [43, 142–144], pseudo-Zernike momenti [42], itd. Treba istaći da, iako su globalne i robustne u različitim zadacima analize, metode zasnovane na oblasti oblika često uključuju intenzivna i prilično zahtevna računanja, i ne obezbeđuju jasnu razliku između sličnih objekata [101]. Međutim, u nekim zadacima analize objekta informacije o granici oblika mogu biti od značajnije od informacija dostupnih iz cele oblasti oblika. Ovo proizilazi iz činjenice da većina ljudi može jedinstveno opisati i identifikovati oblik koristeći njegovu granicu. Kao takvi, obično su efikasniji i lakši za karakterizaciju oblika, u poređenju sa tehnikama zasnovanim na celoj oblasti oblika [101]. Međutim, ove tehnike imaju određena ograničenja koja mogu značajno smanjiti njihovu korisnost i primenljivost. Naime, oni su generalno osetljivi na šum i različite nivoe promene oblika koji su posledica različitih efekata, kao što su, na primer, promena ugla iz kojeg se objekat posmatra, prisustvo različitih prepreka u sceni, poput drveća, zgrada ili vozila koja delimično ili u potpunosti preklapaju oblik koji razmatramo, itd. Pored toga, ove metode su prilično osetljive ako granica oblika nije kompletna ili ako se sam oblik sastoji od više disjunktnih oblasti ili rupa [180]. Do sada su uvedene različite tehnike zasnovane na granici oblika i obično uključuju signaturu oblika [50, 51, 81, 140], Furijeove deskriptore [12, 82, 140, 164, 178], talasiće deskriptore [173, 178], zakrivljenost na višestrukim skalama [101], tehnike na višestrukim skalama [3, 46, 123, 140], kompleksne mreže [7–9, 64, 116], itd.

Drugi kriterijum za klasifikaciju alata za analizu oblika proučavanih u tezi odnosi se na to da li je rezultat njihove primene predstavljen u obliku skalara ili vektora, ili u obliku druge slike oblika ili grafa [94]. U vezi s tim, razlikujemo *tehnike skalarne transformacije* koje transformišu, odnosno preslikavaju sliku koja sadrži oblik u skup skalara ili vektora, i *tehnike transformacije domena* koje transformišu domen slike oblika u drugu odgovarajuću sliku ili graf. U vezi prve grupe tehnika analize, do sada su razvijeni različiti pristupi čiji je cilj numerička karakterizacija oblika. Ovo proizlazi iz činjenice da oblik, kao jedan od osnovnih atributa objekta, ima nekoliko različitih karakteristika (npr. geometrijske, topološke ili njihove odgovarajuće kombinacije) koja se mogu numerički oceniti. Najčešći pristup u zadacima ovog tipa jeste proučavanje određene topološke ili geometrijske karakteristike datog oblika, pogodne za numeričku karakterizaciju (npr. kompaktnost, konveksnost, eliptičnost, izduženost, kvadratnost, itd.), takođe nazvan *deskriptor oblika*, a zatim osmisliti metodu, odnosno *meru oblika*, koja ocenjuje u kom obimu razmatrani oblik zadovoljava ovo svojstvo. Poželjan, ali ne i suštinski zahtev jeste da ponašanje tako osmišljene mere bude intuitivno jasno i nedvosmisleno što je moguće više, jer se tada njeno ponašanje može relativno lako razumeti i predvideti unapred. Glavna ideja je dodeliti nekoliko numeričkih karakteristika, izračunatih iz deskriptora oblika, objektima koji se razmatraju, a zatim ih koristiti kao komponente odgovarajućeg vektora karakteristika dodeljenog svakom posmatranom objektu. To nam omogućava da se određeni zadaci analize objekata, koji se zasnivaju na njihovom upoređivanju, prepoznavanju,

podudaranju ili klasifikaciji, mogu izvoditi u odgovarajućem vektorskom prostoru karakteristika koristeći odabranu metriku, a ne u prostoru objekata. Ovaj pristup se zatim može lako proširiti na različite zadatke analize objekata koji se zasnivaju na obliku, a koji se izvode na računarima. Imajući to u vidu, prirodno je očekivati da veća dimenzija vektora karakteristika može obezbediti veću razliku između razmatranih objekata. Shodno tome, zbog stalnih zahteva za što više numeričkih karakteristika oblika, sve je veća potreba za proučavanjem novih deskriptora oblika, kao i za dizajniranjem novih metoda za merenje već postojećih deskriptora oblika. Ovo poslednje proizilazi iz činjenice da, zbog raznolikosti zadataka analize objekata, kao i postojanja različitih baza objekata različite složenosti, ne postoji jedinstvena mera oblika koja se efikasno može primeniti u svakom zadatku analize objekta, s obzirom da mera koja daje dobre rezultate u jednom zadatku ne mora nužno da se ponaša shodno našim očekivanjima u drugom zadatku. S tim u vezi do sada je razvijeno nekoliko deskriptora oblika, uključujući one za koje je već razvijeno nekoliko pratećih mera oblika. Napomenimo samo nekoliko: konveksnost [86, 113], kompaktnost [88, 127, 128], eliptičnost [34, 160], izduženost [48], kvadratnost [122], vijugavost [28, 76, 114], trougaonost [120], itd. Takođe, postoji nekoliko generičkih deskriptora oblika, koji nisu prvobitno uvedeni za merenje određenog atributa oblika, ali koji mogu pružiti odgovarajući vektor karakteristika koji vrlo dobro može opisati razmatrani oblik. Ovi deskriptori u većini primena nemaju jasnu geometrijsku ili topološku interpretaciju, i obično se uvode kao rezultat nametanja određenih zahteva koje svaki dobro definisani deskriptor oblika treba da zadovolji. Među njima imamo invarijante oblika zasnovane na momentima [60, 66, 153], Furijeove deskriptore [51, 164, 169], Furijeove moment invarijante [165], itd.

Sa druge strane, *tehnike transformacije domena* obezbeđuju reprezentaciju slike originalnog oblika u obliku druge slike (npr. mapa rastojanja [53], transformacija medijalne ose [6, 16], konveksni omotač oblika [152]) ili grafa [7–9, 64, 116], i obično se koristi za strukturalno opisivanje i karakterizaciju objekata [30, 31]. Međutim, tako dobijena reprezentacija oblika se takođe može posmatrati međukorakom koji prethodi numeričkoj karakterizaciji samog oblika. Na ovaj način se izdvajanje karakteristika oblika vrši indirektno iz transformisanog domena oblika, umesto direktno iz njegovog originalnog domena. To proizlazi iz činjenice da gotovo sve tehnike analize oblika, u jednom koraku njihove implementacije, transformišu informacije o obliku zasnovane na slici u skup brojeva ili vektora koji numerički opisuju sam oblik [157]. U tom smislu, mapa rastojanja, graf, odnosno mreža, konveksni omotač ili medijalna osa oblika mogu se koristiti kao međurezultat za alate analize objekata koji pretpostavljaju da je ova vrsta transformacije slike prethodno izvršena. Primeri su brojni, a ovde pominjemo samo nekoliko: merenje rastojanja između objekata na slici, pronalaženje najkraćeg puta između dve tačke u prisustvu prepreka, izračunavanje Furijeovih ili talasića deskriptora, računanje broja objekata prikazanih na slici, itd. Takođe, nekoliko ovako definisanih zadataka obrade slike posebno je razmatrano

i ilustrovano u posebnoj sekciji na kraju Poglavlja 3.

Treći kriterijum klasifikacije tehnika analize oblika odnosi se na pitanje očuvanja informacija o obliku koji se analizira. U zavisnosti od toga da li se razmatrani oblik može jednoznačno ili sa kontrolisanom preciznošću rekonstruisati iz odgovarajuće reprezentacije ili određenog broja deskriptora, razlikujemo metode koje očuvavaju i metode koje ne očuvavaju informacije o obliku. Tako, na primer, imamo da većina već pomenutih deskriptora oblika zasnovanih na geometrijskim ili topološkim karakteristikama (na primer, kompaktnost, izduženost, konveksnost, kvadratnost, itd.) ne očuvavaju informacije o obliku koji analiziraju, te se stoga originalni oblik iz njih ne može rekonstruisati sa kontrolisanom preciznošću. Ovo poslednje sledi iz činjenice da postoje različiti oblici koji imaju iste pridružene mere ovih deskriptora pa je na osnovu njih nemoguće rekonstruisati sam oblik. Sa druge strane, disk kao oblik se može jedinstveno rekonstruisati iz momenata nultog i prvog reda. To proizlazi iz činjenice da je pridruženi moment nultog reda jednak površini diska, dok momenti prvog reda normalizovani površinom oblika određuju koordinate centra diska. Pored toga, postoje reprezentacije oblika koje omogućavaju njegovu rekonstrukciju ukoliko se posmatraju samo određene klase oblika. U vezi s tim, imamo da signatura oblika kao jednodimenzionalna funkcija, koji predstavlja dvodimenzionalni oblik, omogućava njegovu rekonstrukciju u zavisnosti od njene definicije i svojstava oblika na koji se primenjuje. Na primer, u slučaju objekata koji su zvezdasti, signatura zasnovana na rastojanju od centroida oblika omogućava njegovu jedinstvenu rekonstrukciju u idealnom, neprekidnom, i bez prustva šuma, slučaju. Slično se odnosi i na signaturu poprečnog preseka koja omogućava željenu rekonstrukciju ako i samo ako je razmatrani oblik simetričan. Međutim, u slučaju signature zasnovane na kompleksnim koordinatama, odnosno funkciji položaja, graničnih tačaka, ili funkciji ugla tangente određenog u svakoj tački granice oblika, omogućena je jedinstvena rekonstrukcija oblika bez ikakvih dodatnih zahteva ili ograničenja [78].

Pored toga, trebalo bi napomenuti da tradicionalni pristup zadacima analize objekata podrazumeva da je originalna slika prethodno jasno i nedvosmisleno segmentirana u procesu jasne, odnosno oštre segmentacije slike, nakon čega sledi analiza izdvojenih komponenti. Shodno tome, segmentacija slike kao proces razdvajanja originalne slike na jasno definisane oblasti, odnosno komponente koje se sastoji od tačaka koje dele određene zajedničke karakteristike (npr. prostorne, vizuelne, geometrijske, topološke, itd.) jeste od posebne važnosti za kvalitet tehnika koje se koriste u sledećim koracima analize objekata. U tom smislu, suštinsko pitanje koje se u tim zadacima postavlja jeste kako i po kojim kriterijumima se može doneti odluka o tome koje tačke slike pripadaju objektima koji se nalaze na slici. Donošenje takve odluke obično se zasniva na jasnom, nedvosmislenom, dualnom, dvovalentnom, binarnom, da-ne ili tačno-netačno, i ništa između toga, rezonovanju i odlučivanju. Na ovaj način se originalna slika transformiše u oštru (to jest binarnu, dvovalentnu, odnosno crno-belu) sliku gde se tačkama objekta pridružuje vrednost 1,

odnosno nivo bele boje, zasnovan na odgovoru da, tačno, pripada ili jeste član objekta, dok se tačkama koje ne pripadaju obliku pridružuje vrednost 0, odnosno nivo crne boje, shodno odgovoru ne, netačno, ne pripada ili nije član objekta. Međutim, zbog postojanja različitih izvora nesigurnosti i nepreciznosti koji mogu nastati pri radu sa slikama, a do kojih dolazi pre svega usled nesavršenosti uređaja za snimanje slike (npr. ograničene rezolucije, grešaka u diskretizaciji, odnosno uzorkovanja neprekidnog domena slike), nepovoljnih uslova snimanja, prisutnosti šuma koji se ne može izbeći u procesu stvaranja slike, primene različitih algoritama za obradu slike (npr. zamućivanje ili izglađivanje slike), itd., nužno dolazi do pojave nesigurnosti i u procesu donošenja odluke u vezi pripadnosti tačke slike određenom objektu. Pri tome, nesigurnost podrazumeva nedostatak važnih i sa aspekata dizajniranja samog modela odlučivanja suštinskih informacija odakle imamo da se osnovne karakteristike i parametri modela ne mogu sa sigurnošću znati, a neophodne odluke deterministički predvideti i doneti. Ovo prvenstveno sledi iz činjenice da, usled gore pomenutih izvora nesigurnosti, granice različitih objekata koji se pojavljuju na slici postaju nejasne ili neodređene, čime se isključuje sigurnost u procesu odlučivanja o pripadnosti tačke slike određenom objektu na slici. U tom smislu, sasvim je jasno da model odlučivanja zasnovan na donošenju jasne i nedvosmislene odluke nije dovoljno obučen i ne pruža dovoljnu fleksibilnost u cilju odgovarajućeg opisivanja i tretmana ove vrste nesigurnosti. Kao rezultat toga, imamo da ovako definisani model donošenja odluka, zasnovan na principu dihotomije, ne odgovara adekvatno izazovima koji potiču od nepostojanja oštro definisanih granica objekata ili granica između različitih objekata na slici. Shodno tome, model zasnovan na donošenju odluka, praćen principom višemanje, a ne dihotomije da-ne ili tačno-netačno, mnogo je poželjniji i značajniji u zadacima tretiranja ove vrste nesigurnosti, omogućavajući da tačka slike može biti član objekta u određenom stepenu. Na osnovu toga, pripadnost tačke određenom objektu na slici može se izraziti u obliku iskaza čija istinitost ne mora biti apsolutna, isključiva, dvovalentna, odnosno binarna, i može se izraziti stepenom u rasponu od 0 do 1, zavisno od stepena njegove istinitosti. Ova vrsta nesigurnosti koja se odnosi na odsustvo oštro definisanih kriterijuma pripadnosti elementa slike određenom objektu na slici, i koja je data na postepen a ne na apsolutni način, opisujući u kom stepenu je neki iskaz tačan, naziva se *rasplinutost*. U tom smislu, umesto primene binarne segmentacije koja rezultira dvo-vrednosnom, odnosno binarnom slikom, *rasplinuta ili fazi segmentacija*, izvedena na način da tačke mogu da pripadaju nekom objektu u određenom stepenu, transformiše originalnu sliku u sivu sliku, koja se sastoji od tačaka objekta kojima je dodeljen intenzitet 1 ili nivo bele boje, tačaka koje ne pripadaju objektu i kojima je dodeljen intenzitet 0 ili nivo crne boje, i tačaka koje delimično pripadaju objektu i kojima su dodeljene nivoi sive boje između crne i bele boje, u skladu sa njihovom delimičnom pripadanošću objektima na slici. Shodno tome, fazi segmentacija je posebno važna kada informacije koje se odnose na objekte prisutne u originalnoj slici moraju biti sačuvane što je duže moguće u segmentiranoj

slici, što može biti od posebnog značaja u narednim koracima analize. Ovo proizlazi iz činjenice da ova vrsta segmentacije slike smanjuje rizik donošenja pogrešne jasne odluke o pripadnosti tačke objektu u ovom ranom koraku analize, što potencijalno može dovesti do očuvanja značajne količine važnih informacija o objektima koji su prisutni u originalnoj slici.

Međutim, s obzirom da su objekti koje bi trebalo analizirati obično dostupni kao već digitalizovani, ili se u nekom koraku njihove analize moraju digitalizovati, od posebnog značaja u zadacima analize objekata jesu fazi segmentirani objekti definisani na digitalizovanom domenu originalne slike. Konkretno, bavićemo se proučavanjem određene vrste fazi digitalne reprezentacije objekta koja je zasnovana na delimičnoj pokrivenosti elemenata slike neprekidnim objektom prisutnim u originalnoj slici. Takav model digitalizacije generiše reprezentaciju objekta sa nivoima sive boje, koja se takođe naziva *reprezentacija pokrivenosti*, pri čemu je nivo sive boje pridružen svakom elementu slike proporcionalan njegovoj relativnoj pokrivenosti posmatranim neprekidnim objektom. U ovom modelu, vrednosti intenziteta su u opsegu od 0, za elemente koji nemaju presek sa objektom, do 1, za elemente koji su u potpunosti sadržani u objektu, dok su vrednosti između njih dodeljene elementima koji su delimično pokriveni objektom slike i pojavljuju se samo na granici objekta. S tim u vezi, nivo sive boje dodeljen svakom elementu slike može se direktno koristiti za definisanje stepena njegove pripadnosti odgovarajućem digitalnom fazi objektu. Prednosti digitalne reprezentacije slike dobijene kao rezultat fazi segmentacije originalne slike u odnosu na binarnu reprezentaciju mogu se naći u sledećim radovima [26, 146, 147, 150]. Postignuti rezultati takođe podstiču i motivišu dalja istraživanja u raznim zadacima analize i obrade slike, posebno u zadacima poboljšanja performansi različitih tehnika i alata kada je dostupna samo diskretna reprezentacija posmatranog objekta. S tim u vezi, u tezi ćemo predstaviti kako se tako definisana reprezentacija slika može koristiti za poboljšanje performansi ocena ne tako sličnih alata za obradu i analizu slike, kao što je transformacija euklidskim rastojanjima u tri dimenzije (3D EDT) i signatura oblika zasnovana na rastojanju od centroida oblika. Poboljšanja koja će biti predstavljena u tezi odnose se na smanjenu pristranost i varijansu, poboljšanu invarijantnost u odnosu na rotaciju i translaciju, kao i robustnost u prisustvu šuma.

Konačno, uzimajući u obzir sva razmatranja i zapažanja koja su do sada predstavljena u ovom poglavlju, u ovoj tezi predstavićemo nekoliko različitih alata i tehnika analize zasnovanih na obliku koje možemo klasifikovati shodno prethodno navedenim kriterijumima, a u zavisnosti od pristupa koji se koriste za njihovo uvođenje. Tako se, na primer, prema prvom predstavljenom kriterijumu, tehnike uvedene u ovoj tezi mogu klasifikovati kao metode zasnovane na oblasti koje koriste celokupni sadržaj slike koja sadrži objekat od interesa (nove mere oblika šestougaonosti i fazi kvadratnosti, kao i metode za preciznije ocenjivanje EDT u tri dimenzije sa pod-vokselskom preciznošću), ali i tehnike koje se zasnivaju samo na informacijama o granici oblika (kao što je signatura oblika zasnovana na rastojanju od centroida oblika definisana

kao preslikavanje koji svakoj graničnoj tački datog dvodimenzionalnog oblika dodeljuje njeno euklidsko rastojanje od centroida oblika). Što se tiče drugog kriterijuma klasifikacije tehnika i metoda razmatranih u tezi, razlikujemo one koje kao rezultat obezbeđuju numeričku karakterizaciju oblika, odnosno tehnike skalarne transformacije oblika, kao što su nova mera heksagonalnosti i nova mera fazi kvadratnosti obika, kao i novi pristup oceni neprekidne signature oblika zasnovane na rastojanju od centroida oblika koristeći njegovu diskretnu reprezentaciju u slučaju kada je ona jedino dostupna, ali takođe i na one koje pružaju nenumeričku karakterizaciju slike koja sadrži oblik, to jest tehnika transformacije domena slike, kao što je novi pristup ocenjivanja trodimenzionalne transformacije euklidskim rastojanjima sa podvokselskom preciznošću. Konačno, prema trećem kriterijumu klasifikacije metoda analize oblika, razmatrane metode se mogu podelite na one koje omogućavaju rekonstrukciju, sa kontrolisanim nivoom preciznosti, oblika na koji su primenjene, to jest metode očuvanja informacija, kao što je signatura oblika na osnovu rastojanja od centroida, ali i trodimenzionalna EDT, kao i one koji ne očuvaju informacije o obliku, to jest metode neočuvavanja informacija. Među poslednjima su mere heksagonalnosti i fazi kvadratnosti oblika, s obzirom da one ne omogućavaju rekonstrukciju oblika na koji su primenjene, budući da postoji više različitih oblika koji imaju iste preidružene mere. Stoga se rezultati predstavljenih metoda mogu koristiti kao karakteristike predmeta koji se istražuju i kao takvi nalaze svoju primenu u raznim zadacima analize i obrade slike, u prepoznavanju oblika, i identifikaciji i klasifikaciji objekata. Ovo poslednje dolazi iz činjenice da, u većini slučajeva, dobijeni rezultati mogu biti od suštinskog značaja u složenim zadacima kompjuterske vizije koji pretpostavljaju primenu različitih algoritama klasifikacije mašinskog učenja. Pored toga, treba napomenuti da razmatrane tehnike analize oblika nisu razvijene za određeni zadatak ili primenu u analizi objekata zasnovanih na obliku, i kao takve mogu biti od posebne koristi u širokom spektru primena u zadacima obrade slike, zajedno sa zadacima prepoznavanja i klasifikacije objekata. Takođe trebalo bi napomenuti da to ne znači da metode opisane u tezi pružaju najbolje rezultate u svakoj primeni analize objekata. Ovo poslednje proizlazi iz činjenice da različite primene obično zahtevaju proučavanje različitih aspekata analize objekta, čime se u većini slučajeva isključuje mogućnost jedinstveno primenjenog pristupa u svakoj primeni.

Struktura teze

Teza je organizovana u osam poglavlja. Ovde je ukratko dat sadržaj svakog poglavlja. Prvo poglavlje objašnjava potrebu za proučavanjem novih deskriptora oblika, kao i primenu teorije neodređenosti, pre svega teorije fazi skupova u obradi slike i zadacima analize objekata zasnovanih na obliku. Ukratko je predstavljena i motivacija za samu tezu, kao i kratak pregled prethodnih rezultata i izazova koji će biti obuhvaćeni i na odgovarajući način tretirani u tezi.

Poglavlje 2 opisuje teorijske, to jest matematičke koncepte koji će biti od posebnog značaja u jednom delu predstavljenog istraživanja, kao što su teorija fazi skupova i model digitalizacije zasnovan na pokrivenosti kao posebnom slučaju koncepta fazi digitalizacije. U ovom poglavlju sumirani su različiti izazovi koji proizilaze iz procesa segmentacije, odnosno digitalizacije originalne slike, usled prisustva nesigurnosti i neodređenosti u pogledu pripadnosti elemenata slike objektu koji je prisutan na slici, i ilustrovane prednosti primene teorije fazi skupova za rešavanje ovih izazova. Pored toga, u ovom poglavlju predstavljena je digitalna reprezentacija slike zasnovane na delimičnoj prekrivenosti elemenata slike. Vrednost intenziteta pridružena svakom elementu slike u ovoj reprezentaciji biće zatim korišćen u Poglavljima 4 i 6 za poboljšanje performansi ocene transformacije slike euklidskim rastojanjima u tri dimenzije (3D EDT) i signature neprekidnog dvodimenzionalnog oblika zasnovane na rastojanju od centroida. Poglavlje takođe definiše potrebne pojmove koji će biti od posebne važnosti u Poglavlju 7 gde se uvodi novi deskriptor fazi oblika koji definiše u kojoj meri je fazi kvadratan dati fazi oblik.

Poglavlje 3 predstavlja koncept transformacije rastojanjima (DT) u obradi slike sa teorijskog i eksperimentalnog stanovišta, ilustrujući istovremeno većinu DT algoritama za računanje potrebnih rastojanja koji su do sada objavljeni u literaturi. Algoritmi su predstavljeni na sistematski i potpuno prirodan način, počevši od najjednostavnijih do komplikovanijih, pružajući čitaocu mogućnost da bolje razume osnovnu ideju i motivaciju koja stoji iza svakog od razmatranih algoritama. Pored toga, za svaki od posmatranih algoritama predstavljen je odgovarajući pseudo-kod, kao i ilustracija njihove primene, računska složenost, gornje granice odgovarajućih odstupanja od tačne EDT, itd. Primenljivost i prednosti nekoliko proučavanih DT algoritama ilustrovani su u različitim zadacima obrade slike predstavljenim na kraju poglavlja.

Poglavlje 4 opisuje kako se značajno poboljšanje performansi 3D EDT može postići ako se koristi reprezentacija slike zasnovana na pokrivenosti voksela. U takvoj reprezentaciji, vrednost intenziteta pridružena svakom vokselu trodimenzionalne slike koristi se da bi se ocenio položaj objekta unutar voksela. Tako ocenjena pozicija objekta unutar graničnog voksela može se dalje koristiti za poboljšanje ocena različitih 3D EDT algoritama. Ovo poglavlje predstavlja dva nova algoritma za ocenu 3D EDT sa podvokselskom tačnošću. Oba uvedena algoritma su linearna u pogledu vremenske složenosti u odnosu na broj voksela slike. Sva teorijska zapažanja i rezultati koji se odnose na implementaciju predloženih algoritama za ocenu EDT u tri dimenzije detaljno su razrađeni u ovom poglavlju. Eksperimentalna ocena poboljšanja performansi predloženih algoritama u pogledu smanjene varijanse u odnosu na rotaciju i translaciju, kao i povećane tačnosti i preciznosti ocenjenih euklidskih rastojanja, koristeći sintetičke i realne primere, takođe je predstavljena u ovom poglavlju.

Poglavlje 5 uvodi novu meru oblika koja definiše koliko je dati oblik šestougaoni. Uvedena mera ima nekoliko poželjnih svojstava koja bi trebalo da zadovolji svaka dobro definisana mera oblika. Spomenimo samo neke od njih: 1) nova mera heksagonal-

nosti uzima vrednosti iz intervala $(0, 1]$; 2) nova mera dostiže najveću moguću vrednost 1 ako i samo ako je razmatrani oblik šestougao; 3) nova mera oblika je prirodno definisana i teorijski dobro zasnovana; 4) invarijantna je u odnosu na geometrijske transformacije kao što su rotacija, translacija, i skaliranje oblika; i 5) dobijeni rezultati odgovaraju ljudskoj percepciji i intuiciji. Štaviše, nova mera šestougaonosti oblika obezbeđuje takođe nekoliko korisnih posledica čija su svojstva takođe razmatrana teorijski i eksperimentalno ocenjena. Otuda, imamo novu metodu za ocenu orijentacije oblika koja je određena pravcem koji optimizuje novu meru šestougaonosti, kao i novu meru izduženosti oblika koja je definisana kao količnik dužina poluosa odgovarajućeg pridruženog šestougla korišćenog za definisanje nove mere šestougaonosti. Brojni primeri i eksperimenti predstavljeni su u ovom poglavlju kako bi se ilustrovala različita svojstva i ponašanje uvedenih mera oblika. Poglavlje takođe sadrži zadatke klasifikacije koji se izvršavaju na nekoliko poznatih i najčešće razmatranih baza slika u cilju ilustracije efikasnosti i prednosti predloženih mera oblika.

Poglavlje 6 predlaže novi iterativni pristup za ocenu signature oblika na osnovu rastojanja od centroida koristeći informacije o pokrivenosti sadržane u diskretnoj reprezentaciji slike. Predložena metoda ocenjuje u svakoj iteraciji poziciju objekta unutar graničnog piksela, koja se zatim koristi za poboljšanje ocene same signature u sledećoj iteraciji. Predloženi algoritam je jasno i sažeto predstavljen, sa posebnim osvrtom na potrebne pretpostavke i pravce poboljšanja same ocene signature. U ovom poglavlju je takođe predstavljena statistička ocena predloženog algoritma u odnosu na nekoliko do sada proučavanih algoritama za ocenu signature. U tom smislu, dobijeni rezultati potvrđuju značajna poboljšanja u pogledu smanjenja pristranosti i varijanse ocenjene signature, povećanja robustnosti u prisustvu šuma, kao i smanjenja translacione i rotacione varijanse.

Poglavlje 7 sledi isti koncept kao i u Poglavlju 5, ali sada u slučaju nove mere koja ocenjuje stepen do kojeg je neki fazi oblik fazi kvadrat. Nova mera fazi kvadratnosti intuitivno je jasno definisana i teorijski dobro zasnovana, što omogućava da se njeno ponašanje može razumeti i predvideti do određenog nivoa i pre nego što se izvrši određena evaluacija. Sve razmatrane karakteristike nove mere fazi oblika teorijski su potvrđene i ocenjene kroz različite eksperimente i primere. Nova mera pokazala se vrlo korisnom u raznim zadacima klasifikacije objekata koristeći samo nekoliko prilično jednostavnih deskriptora oblika. Razmatrani eksperimenti klasifikacije izvedeni su na tri velike poznate baze slika koje su do sada opsežno proučavane od strane drugih autora u različitim proučavanjima. Posebna pažnja u ovom poglavlju posvećena je odgovarajućoj diskusiji i komentarima u vezi sa upoređivanjem postignutih rezultata sa onima dobijenim u Poglavlju 5 uključivanjem nove mere šestougaonosti oblika.

Poglavlje 8 sumira zaključne primedbe teze i daje kratak pregled originalnih doprinosa predstavljenih u samoj tezi. Takođe su navedeni potencijalni pravci i ideje za buduća istraživanja.

Contents

Chapter 1

Introduction

This thesis considers the quantitative aspects of shape attributes that are suitable for numerical characterization, i.e., shape descriptors, and the theory of uncertainty, in the narrower sense the theory of fuzzy sets, with their application in image processing. Given that human visual system understands the shape as one of the important components necessary to perceive and recognize the objects that surround us, then it is not surprising that shape analysis represents one of the very important steps (after an image acquisition, its potential pre-processing, and object detection, respectively image segmentation) in which certain object features are extracted, and then used in the following image analysis steps. This applies not only to certain simple tasks of recognizing geometric shapes, written numbers or printed characters but also to other more complicated computer vision tasks, where the shape analysis represents only one of the necessary steps before a particular artificial intelligence algorithm or machine learning routine is applied. This comes from the fact that in different fields of research and applications, thanks to the rapid development of image acquisition technologies, we have an opportunity to work with a large amount of image-based data. The challenges of working with images generated in this way can be multiple, given that there is a permanent need for their analysis, processing, comparison or classification. In all of these tasks, depending on the applications and requests set against us, it is necessary to identify as well as understand the context of the imaged objects in order to make the content of the image itself understandable. Consequently, as an answer to these challenges, several different approaches, and computing methods have been developed so far, both from theoretical and experimental points of view. In this regard, it should be noted that the importance and benefits of these methods depend primarily on their applicability in different areas of research. Just to mention a few: agriculture [95, 102, 145, 163], astronomy [49, 65], biology [93, 100], botany [41, 77, 84], ecology [11], geography [5, 168], medicine [28, 55, 106, 114], mobile

robots [56, 98], remote sensing [132–134, 177], transportation [14, 52, 62, 126], etc.

In this thesis, we focus our attention on the object characterization based on the shape, given that the shape as one of the elementary visual object features (together with texture and color) has a variety of attributes which can be evaluated numerically, and, consequently, used for object characterization. It is usually presented as a bounded, not necessarily connected, region specifying the spatial extent of the object when its visual characteristics such as texture and color have been excluded from it. As such, it can be identified with a silhouette of the object obtained by its illumination from an infinitely distant source of light [94]. Importance of a shape as a tool for objects analysis and characterization has led to its study in various object analysis tasks and resulted in the extensive literature related to this topic [9, 47, 67, 83, 109, 116, 125, 139, 166]. In the following, we will present several shape-based tools that differ according to a few criteria, which will allow us to classify the approaches considered in this thesis appropriately. In this respect, over the years, different approaches have been proposed to analyze shapes and they are mainly classified into *region-based* and *boundary-based* approaches [94]. The main distinction among these approaches is based on how features are extracted from the shape [167, 181]. In that context, region-based techniques are focused on global analysis to extract the features representing the shape itself. As such, they are global and have proven to be useful in working with generic shapes and low-quality image data, and they are robust in the presence of noise as well. Among the most commonly used region-based methods are moment invariants [66, 153, 172], compound image descriptor [87], generic Fourier descriptors (GFDs) [180], Zernike moments [43, 142–144], pseudo-Zernike moments [42], etc. It should be pointed out that although they are global and robust in different analysis tasks, the region-based methods often involve intensive and rather tedious computations, and do not provide a clear distinction among the similar objects [101]. However, in some applications, information about the boundary of shape may be of greater importance than the information available from the shape region. This arises from the fact that most people can uniquely describe and identify a shape using its boundary. As such, they usually tend to be more efficient and easily derived, compared to region-based techniques [101]. However, these techniques have certain limitations that can significantly reduce their usability and applicability. Namely, they are generally sensitive to noise and different levels of variation of shapes resulting from different effects, such as, for example, changing the angle from which the object is viewed, the presence of various obstacles in the scene such as trees, buildings or vehicles that partly or completely overlap the shapes we are considering, etc. Besides, these methods are pretty sensitive if the boundary of a shape is not complete and some points are missing, or if the shape itself is consisting of several disjoint regions or holes [180]. So far, various boundary-based techniques have been introduced and they usually include shape signatures [50, 51, 81, 140], Fourier descriptors (FDs) [12, 82, 140, 164, 178],

wavelet descriptors (WDs) [173, 178], curvature scale space (CSS) [101], multiscale techniques [3, 46, 123, 140], complex networks (CNs) [7–9, 64, 116], etc.

Another criterion to classify shape analysis tools studied in this thesis is based on whether the result of their application is numerical or non-numerical [94]. In this regard, we distinguish *scalar-transform techniques* that transform (i.e., map) an image, containing the shape, into a set of numbers (scalars or vectors) and *space-domain techniques* transforming the image of shape into another image or graph. Regarding the first group of analysis techniques, different approaches have been developed so far aiming at the numerical characterization of the shapes. This follows from the fact that the shape, as one of the elementary object properties, has several various properties (e.g., geometrical, topological, or their combinations) that can be evaluated numerically. The most common approach in these tasks is to study a certain topological or geometrical feature of a given shape, suitable for numerical characterization (e.g., compactness, convexity, ellipticity, elongation, squareness, etc), also called *shape descriptor*, and then to devise a method, resp. *shape measure*, that evaluate to what extent the shape considered satisfies this property. A desirable, but not essential requirement is that the behaviour of such a devised method is intuitively clear and unambiguous as much as possible since its behaviour is then relatively easy to understand and predict in advance, even before a specific analysis task is performed. The main idea is to assign several numerical features, computed from the shape descriptors, to object under consideration, and then to use them as the components of the corresponding feature vector assigned to each object observed. This enables us that certain object analysis tasks, which are based on their comparing, recognizing, matching, or classifying, can be performed in the corresponding feature vector space using chosen metric rather than in the object space. This approach can then be easily extended to various shape-based object analysis tasks performed on computers. Taking this into account, it is natural to expect that a larger dimension of the feature vector may provide greater distinction between the objects under consideration. Consequently, due to the permanent demand for more numerical characterizations, there is an increasing need to study new shape descriptors, as well as to design new methods for measuring already existing shape features. The latter results from the fact that, due to the diversity of object analysis tasks as well as the existence of different image databases of objects of varying complexity, there is no single shape measure applying efficiently to every object-based application, since a measure performing well in one application does not necessarily produce results according to our expectations in another application. In this regard, several shape descriptors have been developed so far, including those for which several measures have already been designed. Just to mention a few: convexity [86, 113], compactness [88, 127, 128], ellipticity [34, 160], elongation [48], squareness [122], tortuosity [28, 76, 114], triangularity [120], etc. Also, there exist several generic shape descriptors, not originally introduced to measure particular shape attribute, but which can provide corresponding feature vector

describing the considered shape very well. These descriptors in most applications do not have a clear geometrical or topological interpretation, and they are usually introduced as a result of imposing certain requirements that each well-defined shape descriptor should satisfy. Some of them are moment invariants [60, 66, 153], Fourier descriptors (FDs) [51, 164, 169], Fourier moment invariants [165], etc.

On the other hand, *space-domain techniques* produce a non-numerical representation of the original shape image given in the form of another image (e.g., distance map [53], medial axis transform (MAT) [6, 16], convex hull [152]) or graph [7–9, 64, 116], and is usually used for structural object description and characterization [30, 31]. However, such an obtained non-numerical shape representation can also be considered as an intermediate step preceding numerical characterization of shape. In this way, shape feature extraction is performed indirectly from its transformed domain, instead of directly from the original one. This follows from the fact that virtually all the shape analysis techniques, at one step of their implementations, transform image-based information into a set of numbers describing the shape itself [157]. In this respect, distance map, graph, respectively network, convex hull, or skeleton can then be used as an input for object-based analysis tools that assume that this type of image transformation has been previously performed. Examples are numerous and herein we mention only a few: measuring the distances between the objects, finding the shortest path between two points among the obstacles, computation of Fourier or wavelet descriptors, computing the number of objects shown in the image, etc. Some of these image processing tasks will be particularly considered and illustrated in a separate section at the end of Chapter 3.

The third criterion for classifying the shape analysis techniques is based on preservation information. Depending on whether the shape considered can be reconstructed with a controllable precision from its corresponding representation or from a certain number of descriptors, we distinguish between information-preserving and non-preserving methods. Thus, for example, we have that most already mentioned shape descriptors based on geometrical or topological characteristics (e.g., compactness, elongation, convexity, squareness, etc.) do not preserve information about the shape they analyze, and therefore the original image shape cannot be reconstructed from them, due to existence of different shapes that have the same assigned measures of these descriptors. On the other hand, the shape of disc can be uniquely reconstructed from its zeroth- and first-order moments. This comes from the fact that the associated moment of zeroth-order amounts the area of the disc, while the first-order moments normalized by the shape area determine the coordinates of the disc center. Besides, there exist shape representations that allow its reconstruction if only certain classes or types of shapes are considered. In this regard, we have that the signature of shape as a one-dimensional function, representing the two-dimensional shape, enables its reconstruction depending on its definition and the properties of the shape to which it is applied. For example, in the case of a star-shaped object, the centroid distance sig-

nature enables its unique reconstruction in an ideal, continuous, and noise-free case. The similar refers to the cross-section signature, which allows desired reconstruction only if the shape considered is symmetrical one. However, in the case of the signature based on the complex coordinates, i.e., position function, or tangent-angle function, a unique reconstruction of shapes is enabled without any additional requirements or restrictions [78].

Besides, it should be mentioned that the traditional approach to the tasks of object analysis implies that the original image is previously clearly and unequivocally segmented in the process of image segmentation, followed by the analysis of the extracted objects. Consequently, image segmentation as a process of separating the original image into clearly defined regions, consisting of the points that share certain common characteristics (e.g., spatial, visual, geometrical, topological, etc.), is of utmost importance for the quality of the techniques used in the following analysis steps. In this respect, an essential question arising is how and by what criteria we can make a decision on what image points belong to an object present in the image. Usually, making such a decision is based on a clear, unambiguous, dual, bivalent, yes-no, or true-false, and nothing in between, reasoning and judgment. In this way, the original image is transformed into the crisp (i.e., binary, two-valued, 2-level, or black-white) image where the object points are assigned value 1, resp. white level, based on the answer yes, true, belongs to or is a member of the object, while the non-object points are assigned value 0, resp. black level, related to the answer no, false, does not belong or is not a member of the object. However, due to the existence of various sources of uncertainty and imprecision that may occur when working with images, originated primarily due to imperfection of imaging acquisition devices (e.g., limited resolution, errors in discretization, resp. sampling errors), adverse imaging conditions, presence of noise that cannot be avoided in the imaging acquisition process, application of different image processing algorithms (e.g., image blurring or smoothing), etc., it also necessarily leads to uncertainty in decision-making process regarding the membership of an image point to a particular object. In doing so, uncertainty implies the lack of important and, from the aspects of designing the model of making decisions itself, essential information, contributing that the basic characteristics and parameters of the model cannot be known with certainty, and necessary decisions cannot be deterministically predicted and made. This primarily stems from the fact that, due to above-mentioned sources of uncertainty, the boundaries of different objects appearing in the image become unclear or indefinite, thus excluding the certainty in the process of deciding on the membership of an image point to a particular object. In this regard, it is quite clear that a model based on making a clear and unambiguous decision is not sufficiently trained, and does not provide sufficient flexibility aiming at the appropriate description and treatment of this type of uncertainty. This further implies that the model based on the dichotomy principle does not adequately correspond to reality, given that the conclusions drawn for this model are not relevant and potentially have a

little impact on reality. Consequently, the model based on making decisions followed by the principle of more-less rather than a dichotomy yes-no or true-false is much more desirable and meaningful in the tasks of treating of this type of uncertainty, enabling that image point may be a member of an object to some extent. Based on this, a membership of point to a particular imaged object can be expressed in the form of a statement whose truth need not be absolute, exclusive, bivalent, i.e., binary, and can be expressed in a graded manner ranging from 0 to 1, depending on the degree of its truth. This type of uncertainty relating to the absence of sharply defined criteria of the membership of an image point to an object, and which is given in a graded rather than an absolute manner, describing to what extent a certain statement is true, is called *fuzziness*. In this respect, instead of applying binary segmentation resulted in two-valued (i.e., binary) image, the *fuzzy segmentation*, performed in a manner that the points can belong to an object to a certain level, transforms the original image into the gray-level image, consisting of the object points which are assigned the intensity 1 or white level, non-object points that are assigned intensity 0 or black level, and the points that partially belong to an object, which are assigned the gray-levels in between black and white, according to their partial membership to an imaged object. Consequently, fuzzy segmentation is particularly important when information relating to objects present in the original image needs to be preserved as long as possible in the segmented image, which may be of particular importance in subsequent analysis steps. This follows from the fact that this type of image segmentation reduces the risk of making a wrong clear decision on the membership of image point at this early analysis step, which may potentially lead to the preservation of a large amount of important information about the objects present in the original image.

However, given that the objects we need to analyze are usually available either as already digitized or must be digitized at some stage of their analysis, we are mainly interested in fuzzy segmented objects defined in the digital image space. Specifically, we are dedicated to one particular type of fuzzy digital object representation based on the partial coverage of the image elements by a continuous object present in the original image. Such a digitization model provides a gray-level object representation, also called *coverage representation*, where the gray-level assigned to each image element is proportional to its relative coverage by a continuous imaged object. In this model, intensity values are in the range from 0, for the elements having no intersection with the object, to 1, for the elements that are completely contained into an object, while the values between them are assigned elements which are partially covered by an imaged object and appear only on the object boundary. In this regard, the gray-level assigned to each image element can be directly utilized to define its membership level to the corresponding digital fuzzy object. Advantages of the digital image representation obtained as a result of fuzzy segmentation, relative to the binary digital representation, can be found in [26, 146, 147, 150]. These results also encourage and motivate further research in various image analysis and image processing

tasks, especially in the tasks of improving the performance when only the discrete representation of the object is available. In this respect, we will present how such a defined image representation can be used to improve the performance of the estimates of not so similar image analysis tools such as 3D Euclidean distance transform (3D EDT) and the centroid distance signature of shape. Improvements that will be presented in the thesis refer to decreased bias and variance, improved invariance to rotation and translation, and robustness in the presence of noise as well.

Finally, taking into account all the considerations and observations presented so far in this chapter, in this thesis we will present several different shape-based analysis tools and techniques that differ with respect to three criteria mentioned above, depending on the approaches used to introduce them. Thus, for example, according to the first criterion presented, the techniques introduced in this thesis can be classified as region-based methods that use the entire content of an image containing an object of interest (the new measures of shape hexagonality and fuzzy squareness, as well as methods for sub-voxel precise estimating of the EDT in three dimensions), but also techniques based only on shape boundary information (such as centroid distance signature defined as a mapping which assigns its Euclidean distance from the shape centroid to each boundary point of a given two-dimensional shape assigns). Regarding the second criterion for classifying techniques and methods studied in the thesis, we distinguish those that provide numerical characterization, in the form of numbers or vectors, respectively scalar-transform techniques, such as new measures of shape hexagonality and fuzzy squareness, as well as a new approach to evaluating a continuous centroid distance signature of shape from its discrete representation in the case when it is only available, but also those that provide non-numerical characterization of image-based shape data, i.e., domain-transform techniques, such as a new approach to estimating the three-dimensional Euclidean distance transform estimated with sub-voxel precision. Finally, according to the third classification criterion, the considered methods can be categorized into those that allow the reconstruction of the shape to which they were applied with a controllable level of precision, i.e., information-preserving methods, such as centroid distance signature, but also 3D EDT, as well as those that do not preserve the shape information, i.e., information-non-preserving method. Among the latter are the measures of shape hexagonality and fuzzy squareness, given that the associated shape measures do not provide a distinction between the objects to which they are applied in the sense that there exist multiple different shapes that have the same associated measures, and therefore their reconstruction is not possible. Therefore, the results of our presented methods can be used as features of the objects that are being under the study and, as such, find their application in various tasks of analysis and image processing, pattern recognition, and objects identification and classification. The latter comes from the fact that, in most cases, the results obtained can be essential in the complex tasks of a computer vision that assume the application of diverse machine learning classification algorithms. Besides,

it should be noted that the considered shape analysis techniques have not been developed for a specific shape-based object analysis task or application, and, as such, can be of particular benefits in a wide range of applications in image processing tasks, along with the objects recognition and classification tasks. It should also be noted that this does not mean that the methods described in the thesis perform well in every object-based application. This stems from the fact that various applications usually require a study of different aspects of the object analysis, thus excluding in most cases the possibility of a uniquely applied approach in every application.

1.1 The structure of the thesis

The thesis is organized into eight chapters. A brief survey of the contents of each chapter is given below. The first chapter explains the need for studying new shape descriptors, as well as the application of the theory of uncertainty, primarily the fuzzy set theory in image processing and shape-based object analysis tasks. The motivation for the thesis is also briefly presented, as well as a short survey of the previous results and challenges that will be covered and appropriately treated in this thesis.

Chapter 2 describes the theoretical, i.e., mathematical concepts that will be of particular importance in one part of the presented research such as the fuzzy set theory and coverage digitization model as a special case of fuzzy digitization concept. This chapter summarizes various challenges arising from the process of segmentation, respectively digitization of the original image, due to the presence of uncertainty and vagueness regarding the membership of image elements to an object present in the image, and illustrates advantages of applying the fuzzy set theory to address these challenges. In addition, the chapter presents the digital image representation based on the partial coverage of image elements as a result of coverage digitization of the original image. The intensity value assigned to each image element in this image representation will then be used in Chapter 4 and 6 to improve performances of the estimate both Euclidean distance transform in three dimensions (3D EDT) and centroid distance signature of a continuous two-dimensional shape. The chapter also defines the necessary terms related to the fuzzy set theory that will be of particular importance in Chapter 7 where we introduce a new fuzzy shape descriptor that defines to what extent given fuzzy shape is a fuzzy square.

Chapter 3 presents the concept of distance transforms (DTs) in image processing from both theoretical and experimental point of view, illustrating at the same time most DT algorithms for computing the necessary distances that have been reported so far in the literature. The algorithms are presented in a systematic and completely natural way, starting from the simplest to the more complicated ones, giving the reader ability to better understand the underlying idea and motivation behind each of the algorithms considered. Besides, the corresponding pseudo-code is presented for each of the DT algorithms observed, as well as an illustration of their implementation,

computational complexity, upper bounds of the corresponding deviations from the exact EDT, and so on. The applicability and advantages of several studied DT algorithms are illustrated in the various tasks of image processing presented at the end of the chapter.

Chapter 4 describes how significant improvement in 3D EDT performance can be achieved if a representation of an image based on voxel coverage is used. In such a representation, the intensity value assigned to each voxel of a three-dimensional image is considered to provide an estimate of the object position inside the voxel with sub-voxel precision. Such an estimated object position within the voxel can be further used to improve estimates of different 3D EDT algorithms. This chapter introduces two novel algorithms for estimating sub-voxel accurate 3D EDT. Both new, introduced algorithms are linear in terms of time complexity relative to the number of voxels of the image. All theoretical observations and results related to the implementation of the proposed EDT algorithms in three dimensions are elaborated in detail in this chapter. Experimental evaluation of the performance improvement of the proposed algorithms in terms of reduced variance with respect to rotation and translation, as well as increased accuracy and precision of the estimated Euclidean distances, using both synthetic and real examples, is also presented in this chapter.

Chapter 5 introduces a new measure of shape that defines how much a given shape is hexagonal. The measure introduced has several desirable properties that any well-defined shape measure should satisfy. We will mention only a few of them: 1) the new measure of hexagonality takes values from the interval $(0, 1]$; 2) it is maximized (i.e., the maximal possible value 1 is reached) if and only if the considered shape is a hexagon; 3) the new shape measure is naturally defined and theoretically well-founded; 4) it is invariant with respect to geometrical transformations such as rotation, translation, and scaling); and 5) the results obtained match human perception and intuition. Furthermore, a new measure of shape hexagonality provides several useful consequences whose properties have also been considered theoretically and evaluated experimentally. In this regard, we achieve a novel method for evaluating the shape orientation that is determined by a direction that optimizes new hexagonality measure and a new measure of shape elongation defined as a ratio of the semi-axes lengths of the corresponding associated hexagon used to define a new measure of hexagonality. Numerous examples and experiments are presented in this chapter to provide an illustration of the properties as well as the behaviour of the introduced shape measures. The chapter also contains the classification tasks performed on several well-known and most commonly considered image datasets to illustrate the efficiency and benefits of the proposed shape measures.

Chapter 6 proposes a new iterative approach for evaluating the shape signature based on the centroid distance using the coverage information contained in discrete image representation. The method proposed estimates in each iteration the object position within a boundary pixel, which is then used to improve the signature itself

in the following iteration. The proposed algorithm is clearly and concisely presented, with special reference to the necessary assumptions and directions for improving the signature estimate itself. A statistical evaluation of the proposed algorithm relative to several signature estimation algorithms studied so far is also presented in this chapter. In this regard, the obtained results confirm significant improvements in reducing the bias and variance of the estimated signature, increased robustness in the presence of noise, as well as a reduced translational and rotational variance.

Chapter 7 follows the same concept as in Chapter 5, but now in the case of a new measure that evaluates the degree to which a given fuzzy shape is fuzzy square. The new measure of fuzzy squareness is intuitively clearly defined and theoretically well-founded, resulted that its behaviour can be understood and predicted to a certain level before any evaluation is performed. All the considered features of the new fuzzy shape-based measure have been theoretically treated and evaluated through various experiments and examples. The new measure has proven to be very useful in diverse object classification tasks using only a few fairly simple shape descriptors. The classification experiments discussed have been performed on three large well-known image databases, which have been extensively studied so far in various studies by other authors. Particular attention in this chapter is devoted to appropriate discussion and comments regarding the comparisons of the achieved results with those obtained in Chapter 5 by including the new measure of shape hexagonality.

Chapter 8 summarizes the concluding remarks of the thesis and provides a short review relating to the original contributions presented in this thesis. It also outlines potential directions and ideas for future research.

Chapter 2

Fuzzy set theory

Since its introduction by Zadeh [179], the fuzzy set theory has rapidly developed and explored its applicability in a variety of directions and many disciplines. Examples of applications are diverse and include decision theory, expert systems, information processing, artificial intelligence, robotics, meteorology, agriculture, medicine, pattern recognition, and remote sensing [27, 44, 58, 110, 124, 183]. Image processing and image analysis are also among them. One of the main reasons for this lies in the fact that it provides a fairly powerful apparatus and framework to effectively handle the various levels of uncertainty and vagueness present when dealing with images. The theory of fuzzy sets was originally introduced to generalize the classical notion of set and statement to accommodate them to the uncertainties and ambiguities inherent in human speech, reasoning, and decision making. As such, it is based on the premise that a precise description of many real systems is practically impossible and that non-sharply defined classes play a very important role in human thinking and natural language. In this respect, the fuzzy set theory is a natural framework for handling the challenges in which the source of uncertainty and vagueness is the absence of sharply defined criteria for class membership. This type of uncertainty related to a vague definition of membership criteria is also referred to as imprecision or fuzziness and usually refers to a gradual transition between membership and non-membership rather than abrupt [179].

The causes of imprecision in image processing and image analysis can vary according to their nature and the level of its presence in an image. To mention a few: an imprecise or insufficiently distinct boundary between the different entities in an image, the effects arising from a limited resolution or the presence of noise that can not be avoided in the process of image acquisition, the effects resulting from image processing where, for example, the application of different filtering algorithms can affect a presence of vagueness and imprecision in an image, etc. Fuzziness as a prop-

erty can be recognized both as internal image quality and as an unwanted but natural outcome results from the different imaging conditions [15]. Advantages of representing the objects as the fuzzy sets can be numerous, and they undoubtedly lead to an increased interest in applying the fuzzy set theory in different image processing and object recognition tasks. For example, they are of particular importance if the uncertainty and ambiguity captured by an image need to be taken into account and used as long as possible in the tasks of image data analysis [75]. In this regard, the preservation of fuzziness is usually closely related to the preservation of important image information as well as the components present in the image. Based on this, the concept of fuzzy sets represents a natural and also mathematically well-founded framework for treating and studying such information that results from a fuzziness present in an image. In this chapter, the basic mathematical framework of the theory of fuzzy set will be described, with particular reference to its application in the tasks where uncertainty present in an image is not something undesirable that should be discarded, but potentially additional source of information contributing to the performance improvement of various tools and techniques in image analysis and image processing.

Mathematically speaking, the theory of fuzzy sets represents an extension of the classical set theory where the gradual membership of an element to a set under consideration is allowed. In this regard, let us first recall that in the classical set theory, based on two-valued logic, each element of the reference set is either a member or a nonmember of the observed subset. More formally, let us denote with X the reference set and let S be its arbitrary subset, i.e., $S \subset X$. Then each element x of the reference set X is either a member of the subset S or not a member of S. This is usually described by characteristic function with only two values defining an unambiguous distinction between elements $x \in X$ which are members of S and those which are not. Based on this, the classical subset S is defined by its characteristic function $\chi_S : X \to \{0,1\}$ mapping each $x \in X$ into the two-valued set $\{0,1\}$ as follows:

$$\chi_S(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in S, \\ 0, & \mathbf{x} \notin S. \end{cases} \tag{2.1}$$

Then, for all $\mathbf{x} \in X$, if $\chi_S(\mathbf{x}) = 1$, it follows that $\mathbf{x}$ is a member of S, whereas if $\chi_S(\mathbf{x}) = 0$, then $\mathbf{x}$ is not a member of S.

The characteristic function can be generalized such that the membership values belong to a certain interval, indicating their membership to the observed set in question. In other words, the membership of a given element to a fuzzy set is usually described by a function assigning a value from the unit interval $[0,1]$ to a given element, in accordance with its level of membership to fuzzy set. As expects, larger values imply higher levels of set membership. Such a defined function is also called a *membership function*, whereas the set defined by it is called a *fuzzy set*. Following this, the membership function of a fuzzy set S is a function $\mu_S : X \to [0,1]$, where

$\mu_S(\mathbf{x})$ represents a membership grade of $\mathbf{x} \in X$ to a fuzzy set S. Formally, the fuzzy sets are defined by Zadeh in [179] as follows.

Definition 2.1 [179] *A fuzzy subset S defined on the reference set X is a set of the ordered pairs $S = \{(\mathbf{x}, \mu_S(\mathbf{x})) \mid \mathbf{x} \in X\}$, where $\mu_S : X \to [0,1]$ is a membership function of S, and $\mu_S(\mathbf{x})$ a membership grade of $\mathbf{x} \in X$ to S.*

Three basic operations related to crisp sets such as the standard complement, intersection, and union can also be extended to fuzzy sets. Such generalized operations are also known as the *standard fuzzy set operations*, and herein we will provide their definitions. Before that, the three basic relations among the fuzzy sets are introduced.

Definition 2.2 *A fuzzy subset S is empty, $S \neq \emptyset$, if $\mu_S(\mathbf{x}) = 0$, for all $\mathbf{x} \in X$.*

Definition 2.3 *Two fuzzy subsets S i T are equal, $S = T$, if $\mu_S(\mathbf{x}) = \mu_T(\mathbf{x})$, for all $\mathbf{x} \in X$.*

Definition 2.4 *A fuzzy subset S is a subset of fuzzy set T, $S \subset T$, if $\mu_S(\mathbf{x}) \leq \mu_T(\mathbf{x})$, for all $\mathbf{x} \in X$.*

Definition 2.5 *The standard complement of the fuzzy subset S according to the reference set X is a fuzzy subset $\bar{S}$ given by membership function $\mu_{\bar{S}}(\mathbf{x}) = 1 - \mu_S(\mathbf{x})$, for all $\mathbf{x} \in X$.*

Given two fuzzy subsets S and T, then the *standard intersection*, $S \cap T$, as well as the *standard union*, $S \cup T$, are given as follows.

Definition 2.6 *The standard intersection of fuzzy subsets S and T, $S \cap T$, is defined for all $\mathbf{x} \in X$ with membership function $\mu_{S \cap T}(\mathbf{x}) = \min\{\mu_S(\mathbf{x}), \mu_T(\mathbf{x})\}$.*

Definition 2.7 *The standard union of fuzzy subsets S and T, $S \cup T$, is defined for all $\mathbf{x} \in X$ with membership function $\mu_{S \cup T}(\mathbf{x}) = \max\{\mu_S(\mathbf{x}), \mu_T(\mathbf{x})\}$.*

Further, of particular importance for our research are the crisp subsets related to a fuzzy subset, and which are formally defined as follows.

Definition 2.8 *The α-cut of a fuzzy subset S is a crisp subset S^α defined as $S^\alpha = \{\mathbf{x} \in X | \mu_S(\mathbf{x}) \geq \alpha\}$, for all $\alpha \in (0,1]$.*

Definition 2.9 *The support of a fuzzy subset S is the crisp subset $Supp(S) = \{\mathbf{x} \in X | \mu_S(\mathbf{x}) > 0\}$.*

Definition 2.10 *The core of S is the crisp subset $Core(S) = \{\mathbf{x} \in X | \mu_S(\mathbf{x}) = 1\}$.*

Besides, herein we provide some basic definitions related to the observed fuzzy subset S, originally introduced in [45], that will be needed for better understanding of the research presented in Chapter 6.

Definition 2.11 *A subset $S \subset X$ is a star-shaped relative to a point $\mathbf{x} \in S$, if for all $\mathbf{y} \in S$, a line segment that connects $\mathbf{x}$ and $\mathbf{y}$ is completely contained in S.*

Definition 2.12 *The kernel of a subset S is the set of all $\mathbf{x} \in S$ so that for all $\mathbf{y} \in S$, a line segment connecting $\mathbf{x}$ and $\mathbf{y}$ is contained in S.*

Definition 2.13 *A fuzzy subset S is fuzzy star-shaped relative to $\mathbf{x} \in S$, if, for all $\alpha \in (0,1]$, all its α-cuts are star-shaped relative to $\mathbf{x}$.*

Definition 2.14 *The kernel of a fuzzy star-shaped subset is an intersection of the kernels of all its α-cuts.*

Given that of particular interest for our research is to apply certain shape descriptors and the theory of uncertainty in image processing, the theory of fuzzy sets takes a special place in the research presented in the dissertation itself. This stems from the fact that imaged objects are suitably represented by spatial fuzzy sets defined in the appropriate image space [15], for example, $\mathbb{R}^n$ or $\mathbb{Z}^n$, for continuous or digital images, respectively. The membership value assigned to image point expresses the level of uncertainty in terms of its spatial membership to the observed object, i.e., for any image point $\mathbf{x}$, the membership function $\mu(\mathbf{x})$ represents to what extent that point is a member of fuzzy object. However, bearing in mind that objects available are usually given as already digitized, or if they are not, then at some point of their processing or analysis will have to be digitized, we are interested in digital fuzzy sets defined on the digital grid $X \subset \mathbb{Z}^n$, $n = 2,3$, as the corresponding reference set for $2D$ and $3D$ digital images. Besides, when dealing with digital images only a finite number of gray-levels representing the membership values is available. In this regard, we can assume that there exists r equally distributed values corresponding to every digital image element, according to a level of their belongingness to an image object. Such an integer value r is also called a *membership resolution*, which leads us to the following definition of the *digital fuzzy subset*:

Definition 2.15 *A digital fuzzy subset S defined on the reference set $\mathbb{Z}^n$ represents a set of ordered pairs $S = \{(\mathbf{x}, \mu_S(\mathbf{x})) \mid \mathbf{x} \in \mathbb{Z}^n\}$, where $\mu_S : \mathbb{Z}^n \to \{0,1,\ldots,r\}$ is a membership function of S.*

The digitization of a fuzzy set represents the generalization of the crisp digitization of a given set S, which is formally defined as follows:

Definition 2.16 *For a given continuous set $S \subset \mathbb{R}^n$, inscribed into an integer grid $\mathbb{Z}^n$, the Gauss digitization of S is*

$$\mathcal{D}_G(S) = \left\{ \mathbf{x} \in \mathbb{Z}^n \,\middle|\, \mathbf{x} \in S \right\}. \tag{2.2}$$

The generalization of digitization to fuzzy sets is then formally defined as follows:

Definition 2.17 *For a given continuous fuzzy set $S \subset \mathbb{R}^n$, inscribed into an integer grid $\mathbb{Z}^n$, the r-level digitization of S is*

$$\mathcal{D}_{\mathcal{F}}(S) = \left\{ (\mathbf{x}, \mu_S^r(\mathbf{x})) \,\middle|\, \mathbf{x} \in \mathbb{Z}^n, \mu_S^r(\mathbf{x}) = \lfloor r \cdot \mu_S(\mathbf{x}) \rfloor \right\}, \tag{2.3}$$

where $\lfloor x \rfloor$ denotes the largest integer not greater than x.

Before we are able to apply some of the shape analysis tools, it is necessary that the original image is first segmented, i.e., divided into certain homogeneous components that are present in the image. The process of transforming an input image into the image whose elements sharing similar characteristics (e.g., visual, geometrical, topological) are assigned the same label is called *image segmentation*. As a result of segmentation, the original image is split into homogeneous entities consisting of image elements that have the same assigned label. Usually, the segmentation is performed in a crisp manner where each image element can belong to only one of the components present in an image. Such a generated image segmentation is also referred to as *crisp or binary* segmentation, and it is closely connected with possible loss of important information that is present in the original image. This follows from the crisp decision made on image elements that are partially covered by one or more image components. It is usually achieved using a certain threshold basing on which decision, whether the observed image elements belong to an object or not, is made. However, contrary to the crisp segmentation of an image, the segmentation in which the partial membership of the image elements to more than one image components is allowed, i.e., each image element is assigned values of memberships to more than one image objects, then the resulting segmentation is called a *fuzzy segmentation*. In this case, such an assignment of the membership values, i.e., vector of memberships, reduces the possibility of making a wrong decision with respect to membership of image elements, leading to preserving a potentially large amount of important information contained in the original image. Benefits and importance of fuzzy segmentation are particularly visible in the following step in image analysis consisting of various techniques and procedures used for analyzing image components which are previously obtained in the process of image segmentation. Usually, fuzzy segmentation generates an image where most of the elements are assigned the values close to 0 or 1, and which, as such, can be considered as either background or foreground image elements. The remaining image elements are usually located on the boundary of the object and they are assigned the values between 0 and 1, in accordance with the degree of their membership to the object in question. Such a generated image segmentation is closely connected with the fuzzification of the input image based on the coverage of the image elements by the observed imaged object, where the membership value is specified by the level of its relative coverage by the object itself. In this case, the gray levels assigned to each image element can be directly utilized to

define the membership to the corresponding digital fuzzy object. Such an obtained fuzzy segmentation is called *coverage segmentation*, and it is specified by pure image elements, i.e., elements completely covered by imaged object or background, as well as in-between elements which are partially covered by an imaged object, and which are most likely located on the object boundary. In this thesis, we will present how such a defined image representation can be used to improve the performance of not so similar image analysis tools such as 3D Euclidean distance transform (3D EDT) and the signature of a shape based on the centroid distance function. It should be emphasized as well that these tools are generic ones, not designed for a specific task, and as such, they can be used in various image processing and object analysis tasks. Improvements that will be presented in the thesis refer to decreased bias and variance, improved invariance with respect to rotation and translation, and robustness in the presence of noise as well.

Now, we will provide a formal definition of the coverage representation of the image itself. First of all, it is necessary to mention that the partition of a reference set is commonly defined as a family of disjoint non-empty subsets whose union is equal to the reference set. This definition can be then easily extended to the definition of a fuzzy partition of the reference set.

Definition 2.18 *A fuzzy partition of a reference set X is a family of nonempty fuzzy subsets $\mathcal{P}(X) = \{X_i\}_{i\in I}$ such that $\sum_{i\in I}\mu_{X_i}(\mathbf{x}) = 1$ for all $\mathbf{x} \in X$.*

In that context, we give a general definition of a coverage representation of a given continuous set $S \subset X$ concerning a given partition of a reference set. Then, we proceed with the inclusion of certain restrictions to enable a better adaptation of such a model to tasks that are of interest.

Definition 2.19 *For a given partition $\mathcal{P}(X) = \{X_i\}_{i\in I}$ of the reference set X, the coverage representation of a subset $S \subset X$ with respect to a partition $\mathcal{P}(X)$ is a continuous fuzzy subset $\{(X_i, \mu_S(X_i))|X_i \in \mathcal{P}(X)\}$ where $\mu_S(X_i) = |X_i \cap S|/|X_i|$ and $|S|$ denotes area/volume/Lebegues measure of S.*

Since our research is concerned with improving the performance of certain continuous tools in image processing which are estimated from the discrete (i.e., digital) image representation, we assume that the continuous image space $\mathbb{R}^n$ is the reference set and $\mathcal{P}(\mathbb{R}^n)$ is Voronoi partition of $\mathbb{R}^n$ generated by integer points from $\mathbb{Z}^n$. The Voronoi region generated by integer point $\mathbf{x} \in \mathbb{Z}^n$ is called a *spatial element*, and herein (according to notation from [149]) it will be denoted with $\sigma(\mathbf{x})$, whereas the corresponding membership function $\mu(\sigma(\mathbf{x}))$ with $\alpha(\mathbf{x})$. Therefore, $\sigma(\mathbf{x})$ represents a set of all the points from $\mathbb{R}^n$ that are closer to $\mathbf{x}$ than to any other point in $\mathbb{Z}^n$, in terms of the Euclidean distance. To avoid possible ambiguity, in the case of the points at equal distance from the points in $\mathbb{Z}^n$, we use the convention that the lower and left

edges in each image dimension are contained in the corresponding spatial element. Based on this, Voronoi partition $\mathcal{P}(\mathbb{R}^n)$ of the reference set $\mathbb{R}^n$, specified by points from $\mathbb{Z}^n$, is defined in [149] as a family of the left closed $n-$dimensional unit cubes, centered in $\mathbf{x} \in \mathbb{Z}^n$, i.e.,

$$\mathcal{P}(\mathbb{R}^n) = \{\, \sigma(\mathbf{x}) = \mathbf{x} + [-\frac{1}{2}, \frac{1}{2})^n \,\big|\, \mathbf{x} \in \mathbb{Z}^n \}. \tag{2.4}$$

Given that, according to (2.4), there exists one-to-one correspondence between the integer points $\mathbf{x} \in \mathbb{Z}^n$ and associated spels $\sigma(\mathbf{x})$, we come to the following definition of *coverage digitization* in $\mathbb{R}^n$, which is originally introduced in [149]:

Definition 2.20 *For a given continuous object $S \subset \mathbb{R}^n$, inscribed into an integer grid $\mathbb{Z}^n$, the coverage digitization of S is*

$$\mathcal{D}_{cov}(S) = \big\{ (\mathbf{x}, \alpha(\mathbf{x})) \,\big|\, \mathbf{x} \in \mathbb{Z}^n \big\}, \qquad \alpha(\mathbf{x}) = \frac{|\,\sigma(\mathbf{x}) \cap S|}{|\,\sigma(\mathbf{x})|}. \tag{2.5}$$

It is interesting to mention that in the case when $n = 2$, the corresponding digitization is called a *pixel coverage digitization*, whereas, for $n = 3$, it is also referred to as a *voxel coverage digitization*. However, such a definition of digitization in $\mathbb{R}^n$ implies real coverage values assigned to spels defined on $\mathbb{Z}^n$, and as such it is of little practical importance when working with digital images, given that in such cases there exists a discrete number of limited grey-levels representing the spels coverage. This brings us to the definition of the *quantized coverage digitization*, introduced in [149].

Definition 2.21 *For a given continuous object $S \subset \mathbb{R}^n$, inscribed into an integer grid $\mathbb{Z}^n$, the r-level quantized coverage digitization of S is*

$$\mathcal{D}^r_{cov}(S) = \big\{ (\mathbf{x}, \alpha^r(\mathbf{x})) \,\big|\, \mathbf{x} \in \mathbb{Z}^n \big\}, \qquad \alpha^r(\mathbf{x}) = \frac{1}{r} \Big\lfloor r \frac{|\,\sigma(\mathbf{x}) \cap S|}{|\,\sigma(\mathbf{x})|} + \frac{1}{2} \Big\rfloor. \tag{2.6}$$

It is easy to notice that the set of all the possible coverage values in r-level quantized digitization is $\{0, \frac{1}{r}, \frac{2}{r}, \ldots, \frac{r}{r} = 1\}$, where the integer r representing the number of coverage levels is called a *coverage resolution*. Thus, for $r = 1$, this set corresponds to only two possible gray-levels available for *binary digital image*. However, in the case of some complex objects where the exact coverage of image elements is difficult, or maybe impossible to compute analytically, the appropriate approximation of the corresponding coverage values can be of essential importance. The usual way to achieve this is to split an image element into several sub-elements (e.g., by increasing the image resolution or dilating the object we are considering), and then apply the Gauss digitization to them, as indicated in Definition 2.16. The coverage value is then approximately expressed by the fraction of image sub-elements whose centers are covered by the object itself. This leads us to *r-sampled coverage digitization* of continuous object $S \subset \mathbb{R}^n$, defined in [149], as follows:

Definition 2.22 *For a given continuous object $S \subset \mathbb{R}^n$, inscribed into an integer grid $\mathbb{Z}^n$, the r-sampled coverage digitization of S is*

$$\hat{\mathcal{D}}^r_{cov}(S) = \big\{(\mathbf{x}, \hat{\alpha}^r(\mathbf{x})) \,\big|\, \mathbf{x} \in \mathbb{Z}^n\big\}, \qquad \hat{\alpha}^r(\mathbf{x}) = \frac{|\,\hat{\sigma}^r(\mathbf{x}) \cap S|}{|\,\hat{\sigma}^r(\mathbf{x})|}, \tag{2.7}$$

where r-sampled spel $\hat{\sigma}^r(\mathbf{x})$ represents a set of r^n points belonging to $\sigma(\mathbf{x})$ is defined as

$$\hat{\sigma}^r(\mathbf{x}) = \big\{\sigma(\mathbf{x}) \cap \big(\frac{\mathbf{y} - \delta(r)}{r}\big) \,\big|\, \mathbf{y} \in \mathbb{Z}^n, \delta(r) = \big(\frac{r-1}{2}, \frac{r-1}{2}, \ldots, \frac{r-1}{2}\big)\big\}.$$

Finally, it should be noted that the definition of coverage digitization does not answer the question of how the object of arbitrary complexity can be presented in such an image representation. In the case of a relatively simple, or analytically well-defined continuous subsets, the coverage representation can be computed relying on the initial definition of coverage digitization. On the other side, in the case of more complex real objects, the corresponding coverage segmentation algorithms are used instead to provide coverage image representation. In this context, numerous image segmentation algorithms producing a coverage representation have been suggested in the literature so far. For more details, we refer the reader to works [89, 97, 147, 148]. A formal definition of *coverage image segmentation* is given in [149] as follows.

Definition 2.23 *A coverage segmentation of an image I into m components is a set of ordered pairs*

$$\mathcal{S}_{cov}(I) = \big\{(\mathbf{x}, \alpha(\mathbf{x}) \;\big|\; \mathbf{x} \in I_D \subset \mathbb{Z}^n, \alpha(\mathbf{x})) \in \mathbb{A}_m\big\}, \qquad \alpha_i \approx \frac{|\sigma(\mathbf{x}) \cap S_i|}{|\sigma(\mathbf{x})|}, \tag{2.8}$$

where $S_i \subset \mathbb{R}^n$ is an extent of the i-th image component, and $\mathbb{A}_m$ is a corresponding set of m-component (fuzzy) segmentation vectors, given as

$$\mathbb{A}_m = \big\{\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in [0,1]^m \;\big|\; \sum_{i=1}^{m} \alpha_i = 1\big\}. \tag{2.9}$$

In general case, the continuous subsets S_i are not available, leading that the values of α_k should be estimated only from the data present in the image. Following this, the image segmentation based on the coverage of image elements can also be considered as a coverage partition $\mathcal{P}(I_D)$ of the digital image domain $I_D \subset \mathbb{Z}^n$, specified by a family of fuzzy subsets $\mathcal{S}_{cov}(I)_i = \big\{(\mathbf{x}, \alpha_i(\mathbf{x})) \;\big|\; \mathbf{x} \in I_D\big\}, i = 1, 2, \ldots, m$. Specially, if instead of a set $\mathbb{A}_m$, we consider its subset $\mathcal{A}_m$ consisting of m-component *crisp segmentation vectors*

$$\mathcal{A}_m = \big\{\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \in \{0,1\}^m \;\big|\; \sum_{i=1}^{m} \alpha_i = 1\big\}, \tag{2.10}$$

then the corresponding partition of the digital image domain $I_D \subset \mathbb{Z}^n$, consisting of m disjoint crisp subsets $\mathcal{S}_{crisp}(I)_i = \big\{(\mathbf{x}, \alpha_i(\mathbf{x})) \;\big|\; \mathbf{x} \in I_D,\ \alpha_i(\mathbf{x}) = 1\big\}, i = 1, 2, \ldots, m$.

Chapter 3

Distance transforms

In the various digital image processing tasks, the question of how to measure distances between image elements is of particular importance. Traditionally, the *distance transform* (DT) represents a basic framework for computing these distances. The *distance transform* is usually defined on a digital binary image $I : G \rightarrow \{0,1\}$, $G \subset \mathbb{Z}^n$, consisting of object and background image elements, as the mapping that each element of the image maps to its value of the distance to the nearest element of the object. In other words, the distance transform is a global image transformation converting an input binary image into a gray-level image where the assigned value of each image element corresponds to its distance to the nearest object element. Such a generated image is called a *distance map* of I, and if there is no ambiguity between the generated distance map and the distance transform, they can be identified.

The distance transform can be defined using arbitrary metrics. Among the several ones, the most popular are those based on the Euclidean metric, also called the Euclidean distance transforms (EDTs), whose popularity stems from their desirable properties necessary in many applications. However, although the concept of DT is relatively simple, the computation of EDTs with good efficiency and precision is a rather difficult task. These challenges arise from the fact that the computation of EDT is essentially a global operation, and, as such, computationally very expensive. Based on this, one possibility to overcome these challenges is to propose methods that approximate the global distances by propagating the distances between the neighboring image elements through an image, in a pretty efficient way.

This idea was motivated by the ease of computation and was first published in [117, 118]. Since then many different DT algorithms based on the local distances propagation have been developed in the literature [18, 36, 53, 61]. Such algorithms could differ in terms of their accuracy, computational complexity, the order of scanning the image, possibility for parallelization, etc. Herein, we will present the basic

concept of the DTs from both the theoretical and practical standpoint of view. An overview of most well-known algorithms introduced to date is also provided. The presented algorithms are organized in several sections according to their dimensions and types of the metrics applied to provide a reasonably efficient approximation of EDT. The upper bounds of the difference of each presented DT from the exact EDT are also commented in this chapter following the results presented in [18, 19]. Most of these computations are not presented in detail, even though the results presented can be approximately validated by evaluating distances, in the worst possible case, for a sufficiently large image and then compared them with the corresponding exact Euclidean distances.

One-dimensional DT is presented due to the completeness and better understanding of the basic idea of DT algorithms. Detailed explanations and illustrations of the presented algorithms relating to their implementation, accuracy, and direction of improving the performance are given for two-dimensional algorithms [18, 38, 53]. Regarding the three-dimensional algorithms, there are no special additional challenges, except the need for more resources and efficient presentation of the results obtained [18, 74]. Therefore, in this chapter, we will keep on the two-dimensional case, and then we will follow a similar approach, without considering the details, in the case of three-dimensional DT algorithms. The illustrations and comparisons of the presented algorithms for two-dimensional and three-dimensional DTs are also provided. Given that DT is a generic tool, it is of particular importance in diverse image processing and computer vision tasks. In this chapter, we will also present several examples illustrating the applicability of DT algorithms in such tasks. The latter refers to the applicability of DT in the task of finding the shortest path between two given points among obstacles, with particular reference to the case of robot navigation originally introduced in cite Ilic2018.

3.1 Mathematical background

Even though the idea of *distance transform* is rather simple and intuitively clear, it is convenient to provide its precise theoretical foundation to avoid confusion and ambiguity in working with it. Herein we recall the basic mathematical framework necessary for better understanding of the concept of distance transform.

Definition 3.1 *Let X be an arbitrary non-empty set, and let $d : X \times X \to \mathbb{R}$ be a function which assigns to each pair of elements from X a real number. The function d is a metric on X, if for all $x, y, z \in X$, it holds the following properties:*

1^0 $d(x,y) \geq 0$, *non-negativity,*

2^0 $d(x,y) = 0 \Leftrightarrow x = y$, *separability,*

3^0 $d(x,y) = d(y,x)$, *symmetry,*

4^0 $d(x,z) \leq d(x,y) + d(y,z)$, *triangle inequality.*

The ordered pair (X,d) is called a *metric space*, while a number $d(x,y)$ is called a distance between the elements $x,y \in X$. In short, X is a metric space if a metric d is assumed. The function d is called a *distance* if fulfilling only the property 1^0. If it satisfies the properties $1^0, 2^0$ and 3^0, the distance d is called a *semi-metric*, whereas if it fulfils the properties $1^0, 3^0$ and 4^0, the distance d is called a *pseudo-metric*. To date, various examples of metrics have been defined in the literature. Among them, several ones play an important role in diverse image analysis and image processing tasks. Herein we will mention only some of them.

Example 3.1.1 *The function $d : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ defined as*

$$d(x,y) = |x-y|,\ x,y \in \mathbb{R}, \tag{3.1}$$

is called a natural metric on $\mathbb{R}$.

Example 3.1.2 *The function $d_2 : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, n \in \mathbb{N}$, defined as*

$$d_2(x,y) = \Big(\sum_{i=1}^{n} |x_i - y_i|^2\Big)^{\frac{1}{2}}, \quad x = (x_1,\ldots,x_n),\ y = (y_1,\ldots,y_n) \in \mathbb{R}^n \tag{3.2}$$

is a metric on $\mathbb{R}^n$, which is also called the Euclidean metric on $\mathbb{R}^n$.

Example 3.1.3 *One of the useful generalization of the previous metric defined on $\mathbb{R}^n, n \in \mathbb{N}$, defined as follows:*

$$d_p(x,y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{\frac{1}{p}}, \quad x = (x_1,\ldots,x_n),\ y = (y_1,\ldots,y_n) \in \mathbb{R}^n. \tag{3.3}$$

Several special cases of d_p metric will be of particular importance in the following of this chapter, primarily because of their applicability in various image processing and computer vision tasks. Most interesting cases considered herein are

$$\begin{aligned} &\text{for } p = 1: \quad d_1(x,y) = \sum_{i=1}^{n} |x_i - y_i|, \\ &\text{for } p = 2: \quad d_2(x,y) = \left(\sum_{i=1}^{n} |x_i - y_i|^2\right)^{\frac{1}{2}}, \end{aligned} \tag{3.4}$$

which are also known as a *city block* and *Euclidean metric*, respectively; and for $p = \infty$, when, by definition, we denote with $d_\infty(x,y)$ a metric defined as

$$d_\infty(x,y) = \max_{1 \leq i \leq n} |x_i - y_i|, \tag{3.5}$$

and which is also referred to as a *chessboard metric*. It should be noted that validity for the latter denotation, i.e., when $p = \infty$, is based on the fact that $\lim_{p\to\infty} d_p(x,y) = d_\infty(x,y)$.

Definition 3.2 *Let a distance $d : X \times X \to \mathbb{R}^+ \cup \{0\}$ and non-empty set $S \subset X$ be given. Distance transform (DT) defined on X is a mapping D which assigns to each point $x \in X$ its distance to the set S:*

$$D(x) = \inf_{a\in S} d(x,a), \qquad x \in X. \tag{3.6}$$

From the point of application in image processing tasks of particular interest is to observe the distances (i.e., metrics) defined on some discrete set $G \subset \mathbb{R}^n$. Such a discrete set is also called a *grid* and its elements are then called *grid points*. In a special case, for $G \subset \mathbb{Z}^n$, we also talk about *digital grid* (or *integer lattice*) defined on the reference set $\mathbb{R}^n$. As it has been already mentioned, such a digital grid G also induces a partitioning of $\mathbb{R}^n$ into disjoint regions consisting of the points which are closer to one grid point than to any other points from G. This partitioning of $\mathbb{R}^n$ is called a *Voronoi partition or tessellation* of the reference set $\mathbb{R}^n$ and corresponding regions are also called *Voronoi regions* or *Voronoi tiles*. Also, the corresponding grid points are called *sites, sources* or *influence points*. The Voronoi region associated with an arbitrary grid point is also called a *spel*, short for a *SPatial ELement*. For $n = 2$ and $n = 3$, spatial elements are also called *pixels* (short for picture elements) and *voxels* (short for volume elements), respectively. Besides, a digital grid G also induces a digital distance (i.e., metric) $d : G \times G \to \mathbb{R}$, $G \subset \mathbb{Z}^n$, according to conditions stated in Definition 3.1. In that context, of particular interest are metrics from (3.5) and (3.4) restricted to digital plane $\mathbb{Z}^2$ and digital space $\mathbb{Z}^3$.

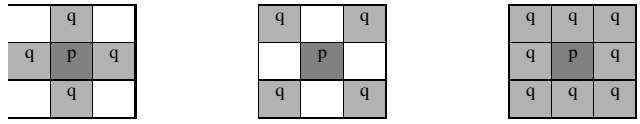
For each pixel $p = (p_x, p_y) \in \mathbb{Z}^2$, there exist four pixels having a common edge with p. These pixels are called *4-neighbors*, and set of all such neighbors is named *4-neighborhood* of pixel p:

$$\mathcal{N}_4(p) = \{(p_x + d_x, p_y + d_y) \mid d_x, d_y \in \{-1,0,1\},\ |d_x| + |d_y| = 1\}. \tag{3.7}$$

In terms of distance, $\mathcal{N}_4(p)$ represents a set of four pixels q that are at a *city block distance* 1 from the pixel p, i.e., $d_1(p,q) = 1$. These four neighbors are connected horizontally and vertically with p, and are therefore called *direct* or *edge neighbors* of pixel p as well (for an illustration, see Fig. 3.1(a)). Besides, four neighboring pixels having a common vertex with pixel p are called *diagonal neighbors*. A set of all diagonal neighbors, denoted as $\mathcal{N}_D(p)$, is called *D-neighborhood* of pixel p:

$$\mathcal{N}_D(p) = \{(p_x + d_x, p_y + d_y) \mid d_x, d_y \in \{-1,0,1\},\ |d_x| + |d_y| = 2\}. \tag{3.8}$$

These four neighbors are also called *indirect* or *vertex neighbors* of pixel p (as given in Fig. 3.1(b)). This brings us to eight neighbors having a common edge or vertex



(a) (b) (c)

Figure 3.1: Illustrations of corresponding neighbors, i.e., neighborhoods, for a given pixel p. (a) Four pixels q represent 4-neighborhood of p, $\mathcal{N}_4(p)$; (b) four pixels q determine D-neighborhood of pixel p, $\mathcal{N}_D(p)$; and (c) eight pixels q represent 8-neighborhood of p, $\mathcal{N}_8(p)$.

with pixel p. Such neighbors are called *8-neighbors* of the pixel p, while a set of all 8-neighbors, $\mathcal{N}_8(p)$, is called 8-*neighborhood* of pixel p:

$$\mathcal{N}_8(p) = \{(p_x + d_x, p_y + d_y) \mid d_x, d_y \in \{-1, 0, 1\},\ \max(|d_x|, |d_y|) = 1\}. \tag{3.9}$$

Following this, we have that 8-neighborhood of p represents a set of all vertical, horizontal and diagonal neighbors of p, i.e., $\mathcal{N}_8(p) = \mathcal{N}_4(p) \cup \mathcal{N}_D(p)$. In terms of distance, $\mathcal{N}_8(p)$ represents a set of eight pixels q that are at a *chessboard distance* 1 from the pixel p, i.e., $d_\infty(p,q) = 1$ (see eight brighter pixels q in Fig. 3.1(c)).

Taking this into account, the *city block* and *chessboard* distances, defined on the digital grid $G \subset \mathbb{Z}^2$, are also known as *4-neighbors* and *8-neighbors* distances. This further implies that these two distances can be denoted for any two grid points $p = (p_x, p_y)$ and $q = (q_x, q_y)$ as follows:

$$\begin{aligned} d_4(p,q) &= |p_x - q_x| + |p_y - q_y|, \\ d_8(p,q) &= max(|p_x - q_x|, |p_y - q_y|). \end{aligned} \tag{3.10}$$

Further, we would like to mention *geodesic* or *path-generated distances* on $\mathbb{Z}^2$. Before that, it is necessary to first define the path between two arbitrary pixels in the digital plane. A *path* $\mathcal{P}(p,q) = (p = p_0, p_1, \ldots, p_n = q)$ from the pixel p to the pixel q in $\mathbb{Z}^2$ is a sequence of distinct pixels $p_i \in \mathbb{Z}^2$, where p_i and p_{i+1} are the neighbors for all $i = 0, \ldots, n-1$. The ordered pairs of two neighbors (p_i, p_{i+1}), $i = 0, \ldots, n-1$, are called the *local steps* along the path. The length of the path $\mathcal{P}(p,q)$ is defined as $|\mathcal{P}(p,q)| = \sum_{i=0}^{n-1} d(p_i, p_{i+1})$, where $d(p_i, p_{i+1})$ is a *local distance* or *weight* of the step (p_i, p_{i+1}) and d is an arbitrary chosen distance (metric) in $\mathbb{Z}^2$. The path $\mathcal{P}(p,q)$ is the *shortest or minimal path* from p to q if its length is minimal among all the existing paths from p to q. Assuming that each pixel is considered as a node on a graph, which is connected to its neighbors depending on the used neighborhood, the digital distance between two pixels is defined in [175] as follows:

Definition 3.3 *The distance between two points p and q is the length of the shortest path connecting p and q in an appropriate graph.*

Following this, we have that the shortest *city block (chessboard) path* from p to q consists of the minimal number of *4-connected (8-connected) neighbors* along the path from p to q. Then the number of neighbors in the minimal *city block (chessboard) path* is called *city block (chessboard) distance* between p and q. It is easy to notice that the diagonal distances are then overestimated (i.e., underestimated) by *city block (i.e., chessboard) metric*, given that the diagonal steps are measured as 2 steps (i.e. 1 step), instead of $\sqrt{2}$ steps. This further implies that the *city-block metric* overestimates, while the *chessboard metric* underestimates the global Euclidean distances. Finally, if the Euclidean distance d_e is observed, then the path with minimal length $|\mathcal{P}(p,q)| = \sum_{i=0}^{n-1} d_e(p_i, p_{i+1})$ is called the *shortest Euclidean path* from p to q, while its appropriate length is called the *Euclidean length of the path* $\mathcal{P}(p,q)$. It is easy to notice that all the shortest Euclidean paths $\mathcal{P}(p,q)$ consist of $(d_4(p,q) - d_8(p,q))$ 8-connected neighbors and $(2 \cdot d_8(p,q) - d_4(p,q))$ 4-connected neighbors, while its corresponding Euclidean length amounts $|\mathcal{P}(p,q)| = (2 \cdot d_8(p,q) - d_4(p,q)) \cdot \mathbf{1} + (d_4(p,q) - d_8(p,q)) \cdot \sqrt{\mathbf{2}}$. Such a defined distance is also called a *quasi-Euclidean distance*, which, following the *triangle inequality*, overestimates the global Euclidean distance, given that the Euclidean distance $d_e(p,q)$ is always shorter than $|\mathcal{P}(p,q)|$ because only horizontal/vertical and diagonal steps have been considered.

Definition 3.4 *Let $G \subset \mathbb{Z}^n, n \in \mathbb{N}$, be a digital grid, $S \subset \mathbb{Z}^n$ an arbitrary non-empty set, and $d : G \times G \to \mathbb{R}$ a given distance defined on G. The distance transform, defined on $G \subset \mathbb{Z}^n$, with respect to S, is a mapping D which assigns to each grid point $p \in G$ its shortest distance to S, in accordance with a given distance d* :

$$D(p) = \min_{a \in S} d(p,a), \ p \in G. \tag{3.11}$$

A given subset S is usually considered to be a binary segmented object, defined on the digital grid G, i.e., $I : G \to \{0,1\}$, $G \subset \mathbb{Z}^n$. From the definition of distance transform (Definition 3.4), it can be easy to devise the simplest, also called *brute-force*, algorithm for computing DT that can be expressed as follows: for each image pixel p, its distance to each object pixel is computed, and minimum of all the computed distances is assigned to pixel p in the distance map D. However, such a computation is essentially a *global operation* which incorporates the global distance minimization for each image pixel independently. In addition, it should be noted that the number of operations required depends not only on the size of the image but also on its content. For example, let suppose that an image of size $n \times n$ consists of k object pixels and $n^2 - k$ background pixels. Given that for each background pixel it is necessary to compare its distance to each object pixel, the number of necessary operations performed

by *brute-force algorithm* is then equal to $Num(k) = k \cdot (n^2 - k)$. After differentiating $Num(k)$ with respect to a number of object pixels, the maximal value is reached for $k = \frac{n^2}{2}$, i.e., where the numbers of the object and background pixels are equal. This further implies that the maximal number of necessary operations is then equal to $Num(\frac{n^2}{2}) = \frac{n^2}{2} \cdot (n^2 - \frac{n^2}{2}) = n^4/2 = \mathcal{O}(n^4)$ [53]. On the other side, $Num(k)$ reaches the minimal value for $k = 1$, as well as for $k = n^2 - 1$, i.e., either if only one object or only one background pixel is given in image. Then the minimal number of necessary operations equals $Num(1) = Num(n^2 - 1) = n^2 - 1 = \Omega(n^2)$. Consequently, depending on the image content, the number of necessary operations is bounded by the best case $\Omega(n^2)$ (lower bound) and the worst case $\mathcal{O}(n^4)$ (upper bound), for an image of size $n \times n$ pixels [53]. Because of this, the computation of DT can be highly costly, and consequently less attractive for practical use. One approach to avoid these challenges is to consider *redundancy* (i.e., *locality*) of the metrics where the global distances are computed by propagating the *local distances* between the neighboring pixels. The observed neighborhood is usually of the size of $3^d, 5^d$ or 7^d elements, where d is image dimension. This idea is based on the following property of metrics on $\mathbb{R}^n$.

Proposition 3.1 *For each point $x \in \mathbb{R}^n$, there exists y from the neighborhood of x, with the same closest influence point.*

Proof 3.1 *Let $z \in \mathbb{R}^n$ be the closest influence point of x, i.e., $x \in \mathcal{VR}(z)$. Since the Voronoi region is an open set, there exists an open neighborhood $B(x,r),\ r > 0$, of points x so that $B(x,r) \subset \sigma(z)$. This establishes the proof, since always exists $y \in B(x,r),\ y \neq x$, which, together with $B(x,r) \subset \sigma(z)$, implies that x and y have the same closest influence point z.* □

In other words, the Voronoi regions defined on the reference set $\mathbb{R}^n$ are always connected sets. If the observed points are restricted to the digital grid, Proposition 3.1 implies that the distance value assigned to each digital point can be computed observing the distance values assigned to its neighbors. This property of metrics is called *regularity*, while metrics satisfying this property are called *regular metrics*.

Definition 3.5 [53] *A metric d is regular if, for every p and q such that $d(p,q) \leq 2$, there exists r, different from p and q, such that $d(p,q) = d(p,r) + d(r,q)$.*

It should be mentioned that most algorithms based on the local distance propagation do not compute the exact EDT since the properties listed in Proposition 3.1 and Definition 3.5 are not satisfied in the case of the Euclidean metric defined on the digital grid. Moreover, if we observe the digital grid $G \subset \mathbb{R}^n$, the closest influence grid point of the observed point is not necessarily the closest influence point of all its (digital) neighbors. This property of discrete Euclidean metric can be expressed as follows:

5	4	5	8	13	20	29
2	1	2	5	10	17	20
1	p_1	1	4	q	10	13
2	1	1	2	4	5	8
4	1	p_2	1	1	2	5
5	2	1	1	p_3	1	4
8	5	4	2	1	2	5

Figure 3.2: Example of disconnected Euclidean Voronoi region using 4-connectivity. Although the pixel q is closer to pixel p_2 than to p_1 and p_3, it achieves a wrong value equals 9 propagated from p_1 or p_3 using 4-connectivity, instead of 8 propagated from p_2. Each color represents the corresponding Voronoi regions. Pixels equidistant to $pv1$ and p_2 are shown in white. The final distance values are then obtained as a square root of the values assigned.

Proposition 3.2 *Discrete Euclidean Voronoi regions are not generally connected subsets.*

Let us consider the discrete Voronoi regions in $\mathbb{Z}^2$ around three given influence pixels p_1, p_2 and p_3, as presented in Fig. 3.2. It can be noticed that the Voronoi region around the pixel p_2 is disconnected relative to 4-connectivity. Therefore, the Voronoi region to which the pixel q belongs cannot be deduced from the Voronoi regions of its 4-neighbors. In other words, if the nearest influence pixel to pixel q is deduced from the four direct neighbors, then the wrong decision can be made given that the pixel p_2 is hidden from q by pixels p_1 and p_3. As a consequence, we have that the discrete Euclidean distance cannot be computed from the Euclidean distances of its 4-neighbors, leading that DT algorithm based on the local distance propagation assigns a wrong distance to q using 4-connectivity. In the case presented, we have that q receives a wrong distance $\sqrt{3^2+0^2}=3$, instead of the exact distance value $\sqrt{2^2+2^2}=\sqrt{8}$. It might be said that this property of a discrete Euclidean metric is the reason why Euclidean DT algorithms were first introduced in the [38], around fifteen years after the first non-Euclidean algorithms that were introduced around 1966 [117]. Distance transform algorithms are traditionally categorized in a way that the local distances are propagated through the image into the *chamfer (i.e., non-Euclidean)* and *vector (i.e., Euclidean)* DTs. In the first group of DT algorithms, the new value of the distance is computed from the distances of its neighbors by adding the appropriate *local distances (or weights)*, contained in the advance chosen neighborhood [18, 19, 22, 25, 154]. Contrary to them, *vector DTs* compute the distance vector from the distance vectors of its neighbors by adding appropriate *local distance vectors*. The necessary distances are then computed from assigned distance vectors [38, 63, 105, 112, 129]. A comparative overview of both distance propagation algorithms can be

0	0	0	1	0	0	0	0	1	0
*	*	*	0	*	*	*	*	0	*
*	*	*	0	1	2	3	4	0	1
3	2	1	0	1	2	2	1	0	1

Figure 3.3: Example of one-dimensional DT algorithm. Four lines from the top to the bottom represent the original one-dimensional image, initialized input image, initialized image after the forward pass, and final image after the backward pass, respectively. The star symbol denotes infinity (i.e., large enough integer).

$F =$ | +1 | 0 | $B =$ | 0 | +1 |

Figure 3.4: Distance masks for one-dimensional DT algorithm

found in [18, 36, 53, 61, 74].

3.2 One-dimensional distance transform

Before we describe the main idea of the algorithms for computing DTs, let first consider one-dimensional binary image (i.e., a string of elements) as given in Fig. 3.3. Ones represent the object elements, while zeros are the background elements. The input binary image is first initialized as follows: *zeros* are assigned to the object elements, while *infinity* (i.e., a sufficiently large integer) is assigned to the background elements. The algorithm consists of two passes over an image. In the first pass, also called *forward pass*, the initial values of the image elements are updated as follows:

$$\begin{aligned} &\text{Forward pass :} \\ &\quad for\ x \leftarrow 2,\ldots,number\_of\_elements \\ &\qquad D^1(x) \leftarrow \min\big(D^0(x), D^1(x-1)+1\big) \end{aligned} \tag{3.12}$$

where $D^0(x)$ and $D^1(x)$ represent the initial and updated value of element x, respectively. The result of the *forward pass* is illustrated in the third line of Fig. 3.3. It can be noticed that each image element is assigned a value equal to its distance to the nearest object element to the left.

In the second pass, also called *backward pass*, the distance values computed in the *forward pass* are updated as follows:

Backward pass :

$$\begin{aligned} &for\ x \leftarrow number\_of\_elements - 1, \ldots, 1 \\ &\quad D^2(x) \leftarrow \min\big(D^1(x), D^2(x+1)+1\big), \end{aligned} \tag{3.13}$$

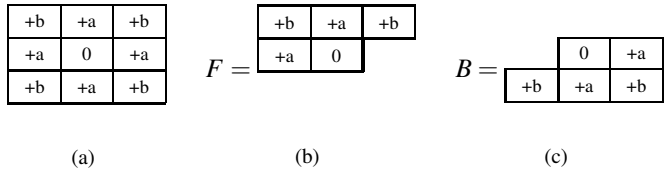
where $D^2(x)$ is updated and final DT value of x. The result of the backward pass is shown in the fourth line of Fig. 3.3. Each image element is assigned the exact value of the distance to the nearest object element.

This algorithm can also be described using the distance masks F and B given in Fig. 3.4. In the forward pass, the zero entry of the forward mask F is placed over the second image element, and the local distances in the mask are added to the image elements below them. The new value of the second image element is then set to the minimum of these sums. The mask F is then moved to the next image element to the right, and its new value is computed in the same manner. This process is repeated until the last element in the image is reached. In a similar way, the backward pass can be described using the backward mask B which is now moved from the right to the left.

Finally, it is worth noting that masks similar to these can be used to provide a better understanding of most DT algorithms. If we understand them once, then we can understand all such defined DT algorithms. The usual procedure can be illustrated in the following way: the local distances in the mask are added to the distance value of image elements covered by the mask, and the new value of the image element (under the center of the mask) is then set to the minimum of all these computed sums.

3.3 Two-dimensional distance transforms

In contrast to one-dimensional distance transform, where the local distances are propagated only along one dimension, two-dimensional propagation-based DT algorithms can be of different properties concerning the accuracy, time complexity, way of implementation, the order of image scanning, etc. In the case of two-dimensional DTs, the original binary image is initialized in a way that the object pixels are set to zeros, while the background pixels are set to infinity (i.e., an integer large enough). The local distances can be propagated through the image both in *parallel* and *sequential* fashion. Following this, we can distinguish *parallel* and *sequential (recursive)* DT algorithms. The underlying idea of propagation-based DT algorithms is to propagate the local distances as waves over the whole image, starting from the pixels initialized by zeros. In the case of the Euclidean distance, the propagation is carried out in circular waves. When the background pixel is reached by a given wave, then the new value is set to the minimum distance from the object. Such a wave propagation can be naturally implemented in a parallel fashion [39, 174].



(a) (b) (c)

Figure 3.5: Distance masks used in 2D algorithms for computing DTs. Mask on the left is used in the parallel computation of DTs, whereas the remaining two on the right are used in the sequential (i.e., recursive) computation of DTs. These two masks are obtained by splitting the left symmetrical mask.

A parallel propagation-based algorithm represents an iterative procedure, where all image pixels are processed in each iteration. The procedure iterates until there are no changes in the distance map during the whole iteration. In the case of 3×3 chosen neighborhood, the parallel algorithms can be described by symmetrical distance mask, as presented in Fig. 3.5(a). During each iteration, the center of the mask is placed over each image pixel, and the local distances are added to the distance values of image pixels below them, which have been computed in the previous iteration. The new distance value of image pixel (below the mask center) is a minimum of all the computed sums. Following this, the parallel algorithm is defined as follows:

$$D^k(x,y) \leftarrow \min_{(i,j)\in mask} \Big(D^{k-1}(x+i,y+j) + d(i,j) \Big), \tag{3.14}$$

where $D^k(x,y)$ is the distance value of image pixel at (x,y) position in the k-th iteration, while $d(i,j)$ is the local distance (i.e., weight) at (i,j) position in the distance mask (centered at $(0,0)$).

Consequently, we have that in each iteration the local distances can be propagated only at a distance equal to the size of the neighborhood considered. Thus, the number of necessary iterations is proportional to the largest distance in the image. The time complexity is then of order $\mathcal{O}(n)$, for an image of size $n \times n$, on the architecture with *one processor per pixel* (i.e., a total of n^2 processors). This leads to time complexity of order $\mathcal{O}(n^3)$, making parallel algorithms more expensive and practically non-efficient.

The equivalent result can be achieved by implementing the *sequential (i.e., recursive)* propagation-based DT algorithms. The first sequential algorithms have been introduced by Rosenfeld and Pflatz [117, 118] with non-Euclidean metrics such as *city block, chessboard* and *octagonal* metrics. However, these metrics provide a rough approximation of the corresponding Euclidean distances, making them less interesting from the aspect of accuracy, but also interesting when the emphasis is on the speed of DT computation. Following this idea, many authors have studied this approach

to improve the approximation of the Euclidean metric [18, 19, 25, 103]. In the sequential DT algorithms, image pixels are processed line by line, from the top left corner to the bottom right corner of the image, and then in opposite direction. The distance value of each image pixel is updated using its current distance value and incremented distance value of previously processed neighbors. The propagation goes in direction from previously processed to the presently processed image pixel. When an image pixel is currently processed, its distance value is compared with the distances assigned to its neighbors incremented by corresponding local distances. Minimum of these computed sums is assigned to current image pixel as its updated distance value. In this fashion, distance values are propagated from each object pixel in the appropriate direction. In the case of 3×3 chosen neighborhood, the distances from *three pixels above and one left* to the current pixel are incremented by appropriate local distances and then propagated in the *forward pass*. In the *backward pass*, the distances from *three pixels below and one right* to the current pixel are propagated in the opposite direction, previously incremented by appropriate local distances. Each pass utilizes only the distance values that have already been updated.

These algorithms can be illustrated using the distance masks which are obtained by splitting the symmetrical parallel distance mask (presented in Fig. 3.5(a)) into the distance masks given in Fig. 3.5(b) and Fig. 3.5(c). In the *forward pass*, the zero entry of the *forward mask* F is positioned over the second image element in the second row of the image, and then the local distance values in the distance mask are summed with the distance values of the image elements below them. Minimum of these sums is assigned to image pixel below the zero entry of the distance mask. The mask F is then moved to the next image element in the same row, and the new distance value is computed in the same manner. This process is repeated from the left to the right until the last element in the second row is reached, and then for each row from the top to the bottom of the image. In the *backward pass*, the zero entry of the *backward mask* B is placed over the image element in the row and column before the last row and column. The new distance value is computed in the same manner as in the forward pass. The mask B is then moved from the right to the left along the same row until the element in the first column is reached. This process is then repeated along each row from the bottom to the top of the image. The pseudo-code of the sequential propagation-based DT algorithm is then given as follows:

Forward pass :

$$for\ y \leftarrow 2, \ldots, number\_of\_rows$$

$$for\ x \leftarrow 2, \ldots, number\_of\_columns$$

$$D^{fwd}(x,y) \leftarrow \min_{(i,j)\in F}\left(D^0(x+i,y+j)+d(i,j)\right)$$

Backward pass :

$$
\begin{aligned}
&for\ y \leftarrow number\_of\_rows - 1, \ldots, 1 \\
&\quad for\ x \leftarrow number\_of\_columns - 1, \ldots, 1 \\
&\qquad D^{bwd}(x,y) \leftarrow \min_{(i,j)\in B}\Big(D^{fwd}(x+i,y+j) + d(i,j)\Big)
\end{aligned}
\tag{3.15}
$$

where D^0 is initialized image, D^{fwd} (D^{bwd}) is distance value of image pixel at (x,y) position in the forward (backward) pass, $d(i,j)$ is the local distance in the distance mask, and *number_of_rows* and *number_of_columns* are the numbers of image rows and columns, respectively.

Based on this, the time complexity of sequential DT algorithms is of order $\mathcal{O}(n^2)$, for an image of size $n \times n$, on the architecture of only one processor [39]. This result makes sequential algorithms much more efficient than parallel ones, requiring *only one processor* for implementation. Besides, parallel and sequential algorithms provide the same final DTs in most cases, with the exception that the sequential algorithms are significantly more efficient than the parallel ones on the sequential architectures. This is the main reason why in our research we are focused on sequential DT algorithms. To illustrate the difference between the parallel and sequential DT algorithms, we present one simple, but also illustrative example.

Example 3.3.1 *The local distances a and b in the distance masks, given in Figure 3.5, represent the distances between the direct and diagonal neighbors, respectively. For different values of local distances, we get different chamfer (i.e., weighted) DTs. E.g., for 4-neighbors DT, we have* $a = 1$ *and* $b = \infty$, *indicating that each sum including b will be ignored from computing the new distance value since only the horizontal/vertical steps are considered. On the other side, for 8-neighbors DT, we have* $a = 1$ *and* $b = 1$, *implying that all local distances (horizontal, vertical and diagonal) are considered. Parallel and sequential algorithms for 8-neighbors (i.e., chessboard) DT to the binary image, i.e., its initialization (given in Fig. 3.6), are illustrated in Fig. 3.7 and Fig. 3.8, respectively. The parallel algorithm is implemented using the distance mask presented in Fig. 3.5(a), while the sequential algorithm is implemented using the distance masks F and B, given in Fig. 3.5(b) and 3.5(c), in the forward and backward pass, respectively. As noticed, the final computed distance map is the same regardless whether the parallel or sequential DT algorithm is applied to compute the required distances. Again, it should be noted that this latter does not have to be true in the general case.*

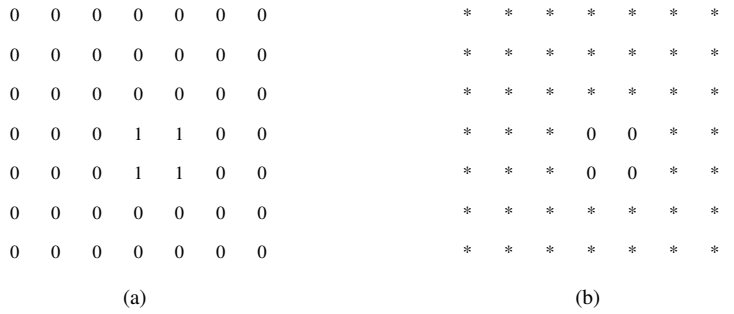


Figure 3.6: Illustration of the first step in chamfer DT computations where the original binary image (ones and zeros represent object and background, respectively) is given in (a), and its initialization (the star symbol denotes infinity, i.e., an integer large enough) is given in (b).

3.3.1 Chamfer distance transforms

Chamfer distance transforms usually refer to algorithms with the local distances a and b set to arbitrarily real positive numbers. In this regard, the 4-neighbors and 8-neighbors DTs are also the chamfer ones with local distances $a = 1$ and $b = \infty$, as well as $a = 1$ and $b = 1$, respectively. However, not all combinations of local distances can give meaningful DTs. To ensure this, the local distances should be selected in a way to meet appropriate constraints. In the case of 3×3 neighborhood, the natural constraints relating the local distances a and b are defined as follows:

$$0 < a < b < 2 \cdot a, \tag{3.16}$$

given that the diagonal step is never shorter than the horizontal (or vertical) step, and is always shorter than one horizontal plus one vertical step. Otherwise, the diagonal step would be shorter than one horizontal plus one vertical step, and as such, it would be ignored. Elaborating this, the *chamfer distance* between two arbitrary pixels $p = (p_x, p_y)$ and $q = (q_x, q_y)$ is then defined as follows:

$$d(p,q) = d_y \cdot b + (d_x - d_y) \cdot a, \tag{3.17}$$

where d_x and d_y are the numbers of horizontal and vertical steps between p and q. Without loss of generality, we can suppose that $0 \leq d_y \leq d_x$. Otherwise, d_x and d_y can change places in (3.17). This follows from the fact that the minimal path between any two pixels which consists of at most two straight-line segments always exists. A proof of the existence of such a minimal path can be found in [103].

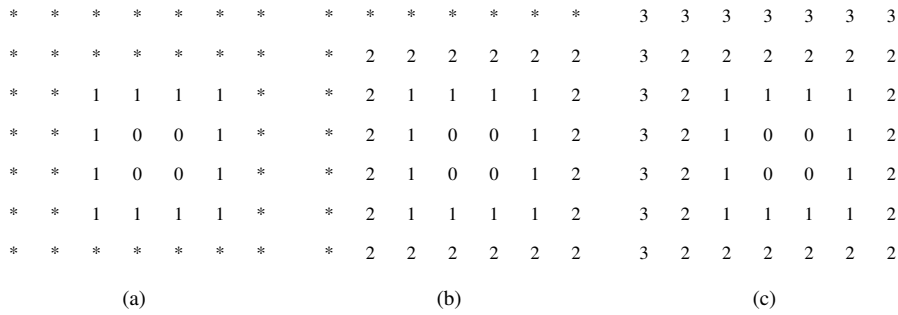


Figure 3.7: Example of parallel chessboard DT algorithm to image given in Fig. 3.6. The parallel algorithm, described by the mask presented in Fig. 3.5(a), is performed through the first (a), second (b) and third iteration (c).

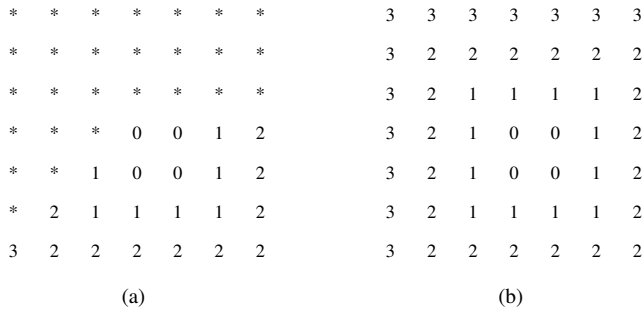


Figure 3.8: Example of sequential chessboard DT algorithm to image given in Fig. 3.6. The sequential algorithm, illustrated by masks in Fig. 3.5(b) and Fig. 3.5(c), is performed using the forward (a) and backward pass (b).

To estimate how well 4-neighbors (i.e., city block) and 8-neighbors (i.e., chessboard) distances approximate the global Euclidean distances, in [18] it has been estimated an upper bound of their differences from the corresponding exact Euclidean distances in the worst possible case. All such computed differences are obtained similarly, so that, without going into details, herein we will illustrate only few of them. For example, the difference of the distance (3.17) from the corresponding Euclidean distance is given as follows:

$$Diff = \sqrt{d_x^2 + d_y^2} - d_y \cdot b - (d_x - d_y) \cdot a. \tag{3.18}$$

In the case of 8-neighbors distance d_8, equality (3.17) becomes $d_8(p,q) = d_x$, while the difference (3.18) reaches the maximal value $(\sqrt{2}-1) \cdot d_x$, for $d_y = d_x$. Consequently, an upper limit of the difference, for an image of size $N \times N$, amounts $(\sqrt{2}-1) \cdot N \approx 0.41 \cdot N$, implying that 8-neighbors distance d_8 always underestimates the corresponding Euclidean distance. However, in the case of 4-neighbors distance d_4, equality (3.17) becomes $d_4(p,q) = d_x + d_y$, while the corresponding difference from the exact Euclidean distance becomes

$$Diff = \sqrt{d_x^2 + d_y^2} - d_x - d_y. \tag{3.19}$$

The absolute value of the difference (3.19) reaches the maximum value $(\sqrt{2}-2) \cdot d_x$ again for $d_y = d_x$. An upper bound of the difference is then equal to $(\sqrt{2}-2) \cdot N \approx -0.59 \cdot N$, indicating that 4-neighbors distance d_4 always overestimates the corresponding Euclidean distance. Also, it can be noticed that 8-neighbors distance provides slightly better result than 4-neighbors distance, as expected.

Further, relying on the main result of [103] (that the shortest path between two pixels consists of only horizontal/vertical and diagonal steps), Montanari has proposed the *quasi-Euclidean* distance assigning the exact Euclidean lengths to the local distances a and b:

$$a = 1, \quad b = \sqrt{2}. \tag{3.20}$$

Such defined distance, also known as *chamfer-Euclidean* distance, always overestimates the Euclidean distance. This result follows from the fact that the *Euclidean length* of the shortest path between two image pixels is always shorter than the corresponding Euclidean distance between them (following the *triangle inequality*). Then, the difference (3.18) becomes

$$Diff = \sqrt{d_x^2 + d_y^2} - d_y \cdot \sqrt{2} - (d_x - d_y), \tag{3.21}$$

which achieves the maximum $(\sqrt{2\sqrt{2}-2}-1) \cdot d_x$, for $d_y = \sqrt{(\sqrt{2}-1)/2} \cdot d_x$. An upper bound of the difference (3.21) is equal to $(\sqrt{2\sqrt{2}-2}-1) \cdot N \approx -0.09 \cdot N$, confirming again that the *quasi-Euclidean distance* always overestimates the Euclidean distance.

However, it can be noticed that all aforementioned local distances a and b do not provide the best possible (i.e., optimal) approximation of the Euclidean distance. Herein, the term optimality usually relates to different criteria for determining the local distances to minimize the corresponding difference from the exact Euclidean distance. One choice for the optimal local distances is to set $a = 1$ and $1 < b < 2$, and then compute an optimal local distance b to minimize the difference defined as

$$Diff = \sqrt{d_x^2 + d_y^2} - d_x - (b-1) \cdot d_y, \tag{3.22}$$

where $0 \leq d_y \leq d_x$. The optimal value is then the value minimizing the function

$$\max(1 - \sqrt{2b - b^2}, |b - \sqrt{2}|),\ 1 < b < 2, \tag{3.23}$$

i.e., the value b representing the solution of the equation

$$1 - \sqrt{2b - b^2} = \sqrt{2} - b, \tag{3.24}$$

that is,

$$b = 1/\sqrt{2} + \sqrt{\sqrt{2} - 1} \approx 1.3507. \tag{3.25}$$

Based on this, an upper limit of the difference (3.22) amounts $1/\sqrt{2} - \sqrt{\sqrt{2} - 1} \cdot N \approx 0.06 \cdot N$, for an image of size $N \times N$. Such a result also confirms that the optimal local distances provide the best possible approximation of the Euclidean metric, among all the already mentioned distances such as 4-neighbors, 8-neighbors, and quasi-Euclidean metrics.

However, for the sake of computational efficiency, in most digital image processing tasks, integer arithmetic is more convenient. This results from the fact that the appropriate integer distances are more preferable if the accent is on the speed and ease of DT computation, aimed at avoiding floating-point operations. Therefore, an appropriate integer approximation of the optimal local distances, so-called *sub-optimal integer distances*, are of particular interest. Usually, the sub-optimal approximations can be obtained by multiplying the optimal local distances with an integer factor and then rounding them to the nearest integer. Thus, for example, in the case of optimal distances $a = 1$ and $b \approx 1.3507$ (3.25), and integer factor 3, we get the following integer distances:

$$a = 3,\ b = 4. \tag{3.26}$$

The final DT is then computed dividing the distances generated by 3. An upper bound of the difference is then equal to $(\sqrt{2} - 4/3) \cdot n \approx 0.08 \cdot n$, for an image of the size of $n \times n$, leading that a greater difference from the EDT is achieved, compared to the one obtained using the corresponding optimal local distances. This is something that can be expected, given that the integer distances are not the exact optimal ones, but their approximations, implying that a greater deviation from the corresponding exact EDT is more expected.

3.3.2 Vector distance transforms

In the previous section, we have noticed that the accuracy of the corresponding chamfer DTs decreases as the size of an image increases. As it has been mentioned, this follows from the fact that the corresponding differences from the exact Euclidean distances are dependent on the image size. To provide a better approximation of the global Euclidean distances, Danielsson [38] has introduced two sequential algorithms for computing EDT, which are performed like chamfer DT algorithms. In such algorithms, instead of the relative distances, the absolute values of the relative coordinates to the nearest object pixel (i.e., *integer distance vectors*) are propagated over an image. The final Euclidean distances are then computed as the *Euclidean lengths* of the corresponding distance vectors assigned to each image pixel.

Two sequential Euclidean distance algorithms, proposed by Danielsson [38], differ concerning the neighborhood observed. The simpler one, *4-neighbors sequential Euclidean distance algorithm (4SED)*, requires checking only *four direct neighbors (horizontal and vertical)* of each image pixel. To improve accuracy, the author proposed using a larger neighborhood for distance vector propagation. This results in another sequential algorithm, also called *8-neighbors sequential Euclidean distance algorithm (8SED)*, which relies on inspecting all *eight closest neighbors* of each image pixel. Consequently, the 8SED algorithm is more numerically complex (i.e., expensive), given that more visited neighbors require more numerical operations.

These algorithms are similar to the corresponding chamfer DT algorithms, with one essential change. In the case of sequential chamfer DT algorithms, described in Section 3.3.1, the relative distances are propagated over an image at propagation angle of 135°, while herein the distance vectors are propagated at an angle of 180°. This is achieved by modifying the raster scanning procedure with two additional scans along each image row, from left to right and from right to left, at each step in the vertical direction. These two additional scans require more computations, given that more comparisons of the distance vectors are needed. Besides, most image pixels are processed and updated more than once until they receive the final distance value.

Following this, in the sequential EDT algorithms, *four passes*, arranged into two *super passes*, over an image, are necessary. In both super passes, an image is inspected both from the left to the right, and from the right to the left. Each pixel, centered at (x,y), is assigned the distance vector $\mathbf{D}(x,y)$, which, at the end of the algorithms, will point out to the nearest object pixel. Contrary to the chamfer DT algorithms, the original binary image is now initialized by two different distance vectors: object pixels are set to vector $\mathbf{0} = (0,0)$, while the remaining image pixels are set to vector $* = (*,*)$ (where $*$ denotes a suitably large integer):

$$\mathbf{D}(x,y) \leftarrow \begin{cases} (0,0), & \text{if } (x,y) \text{ belongs to the object,} \\ (*,*), & \text{otherwise.} \end{cases} \tag{3.27}$$

Sequence of the required operations consists of comparing the distance from the

current pixel (x,y) to the pixel $(x,y)+\mathbf{D}(x,y)$ with the distances from the current pixel to the pixels $(x_{neigh},y_{neigh})+\mathbf{D}(x_{neigh},y_{neigh})$, where $x_{neigh}=x\pm 1$ and $y_{neigh}=y\pm 1$ denote corresponding neighbors of the current pixel. If some of the computed distances is less than the length of the vector assigned to the current pixel, the distance vector $\mathbf{D}(x,y)$ is then replaced by vector $(x_{neigh},y_{neigh})+\mathbf{D}(x_{neigh},y_{neigh})-(x,y)$, where (x_{neigh},y_{neigh}) is chosen to reach the smallest distance from (x,y). The *first super pass* of 4SED algorithm is then defined as follows:

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y),\ \mathbf{D}(x,y-1)+(0,1), \\ \mathbf{D}(x-1,y)+(1,0), \end{array} \right. \tag{3.28}$$

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y), \\ \mathbf{D}(x+1,y)+(1,0), \end{array} \right. \tag{3.29}$$

where passes along each row y go from the left to the right (3.28), and then from the right to the left (3.29). The *second super pass* consists of the similar operations, but in opposite direction, i.e., from the bottom to the top, and along each row y from the right to the left (3.30), and then from the left to the right (3.31):

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y), \\ \mathbf{D}(x,y+1)+(0,1),\ \mathbf{D}(x+1,y)+(1,0), \end{array} \right. \tag{3.30}$$

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y), \\ \mathbf{D}(x-1,y)+(1,0). \end{array} \right. \tag{3.31}$$

The 8SED algorithm is performed similarly as 4SED one, with the difference that more neighboring pixels are inspected. The *first super pass* is given as:

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y), \\ \mathbf{D}(x-1,y-1)+(1,1),\ \mathbf{D}(x,y-1)+(0,1), \\ \mathbf{D}(x+1,y-1)+(1,1),\ \mathbf{D}(x-1,y)+(1,0), \end{array} \right. \tag{3.32}$$

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y), \\ \mathbf{D}(x+1,y)+(1,0), \end{array} \right. \tag{3.33}$$

while the *second super pass* is defined as

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y), \\ \mathbf{D}(x+1,y+1)+(1,1),\ \mathbf{D}(x,y+1)+(0,1), \\ \mathbf{D}(x-1,y+1)+(1,1),\ \mathbf{D}(x+1,y)+(1,0), \end{array} \right. \tag{3.34}$$

$$\mathbf{D}(x,y) \leftarrow \min \left\{ \begin{array}{l} \mathbf{D}(x,y), \\ \mathbf{D}(x-1,y)+(1,0), \end{array} \right. \tag{3.35}$$

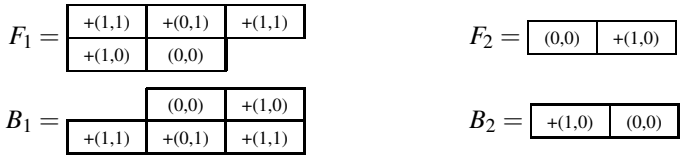


Figure 3.9: Distance masks for two-dimensional vector DT algorithm

where, for the sake of simplicity, in both described algorithms we use notation $\mathbf{v} \leftarrow \min\{\omega \mid \omega \in V\}$, instead of $\mathbf{v} \leftarrow \underset{\omega \in V}{arg\,\min} |\omega|$. The final Euclidean distance map is then computed as the Euclidean length of the distance vector assigned to each image pixel.

It is also worth noting that both described sequential EDT algorithms can be completely described by masks of the relative positions given in Fig. 3.9, where the distance masks for 4SED algorithm are obtained by omitting the distance vectors $(1,1)$. In each pass, the distance masks are moved over an image, and its corresponding distance vectors are then added to the distance vectors assigned to image pixels below them. The distance vector with the minimal Euclidean length is then assigned to image pixel below the center of the mask. In the *first super pass*, the *forward mask* F_1 is moved from the left to the right along the second image row, and then the *forward mask* F_2 is moved from the right to the left along the same image row. These two passes are repeated along each image row, from the top to the bottom of the image. In the *second super pass*, the *backward masks* B_1 and B_2 are moved in the opposite direction (the mask B_1 from the right to the left, and then the mask B_2 from the left to the right). The pseudo-code of both EDT algorithms is given as:

Forward super pass :

$for\ y \leftarrow 2,\ldots,number\_of\_rows$

$for\ x \leftarrow 2,\ldots,number\_of\_columns$ // * the first pass

$$\mathbf{D}(x,y) \leftarrow \min_{(i,j)\in F_1}(\mathbf{D}(x+i,y+j)+\mathbf{v}(i,j))$$

$for\ x \leftarrow number\_of\_columns-1,\ldots,1$ // * the second pass

$$\mathbf{D}(x,y) \leftarrow \min_{(i,j)\in F_2}(\mathbf{D}(x+i,y+j)+\mathbf{v}(i,j))$$

Backward super pass :

$$
\begin{aligned}
&for\ y \leftarrow number\_of\_rows - 1, \ldots, 1 \\
&\quad for\ x \leftarrow number\_of\_columns - 1, \ldots, 1\ //* \text{ the third pass} \\
&\qquad \mathbf{D}(x,y) \leftarrow \min_{(i,j)\in B_1} (\mathbf{D}(x+i, y+j) + \mathbf{v}(i,j)) \\
&\quad for\ x \leftarrow 2, \ldots, number\_of\_columns\ //* \text{ the fourth pass} \\
&\qquad \mathbf{D}(x,y) \leftarrow \min_{(i,j)\in B_2} (\mathbf{D}(x+i, y+j) + \mathbf{v}(i,j))
\end{aligned}
\tag{3.36}
$$

where $\mathbf{D}(x,y)$ is the distance vector at (x,y) image position, $\mathbf{v}(i,j)$ is a local distance vector at (i,j) position in the distance mask (centered in $(0,0)$), while *number_of_rows* and *number_of_columns* are the number of image rows and columns, respectively.

Advantages of 4SED and 8SED algorithms, compared to chamfer DT algorithms is visible by observing the maximum difference (i.e., error) from the exact Euclidean DT. In [38] the author has shown that the maximum difference from the exact EDT, for 4SED and 8SED algorithms, does not depend on the image size, and amounts **0.29** and **0.09** of an image pixel, respectively, with most image pixels having the exact Euclidean distance. Although these algorithms do not always provide the exact Euclidean DT, these errors (less than half of the pixel) make these algorithms very popular and interesting for further research. E.g., in [176] the author introduced two refined versions of 4SED and 8SED algorithms, where the signed relative coordinates to the nearest object pixel are propagated over an image, instead of the absolute ones. The simpler one, called the *4-neighbors signed SED algorithm (4SSED)*, represents the signed extension of the 4SED algorithm, while the second one, called *8-neighbors signed SED algorithm (8SSED)*, is more accurate and represents the signed extension of 8SED algorithm. These algorithms produce the signed distance map, where each pixel is assigned the signed distances to the nearest object pixel along both coordinate (i.e., image) axes. Given that the coordinate distances are assigned to each image pixel, together with their orientation, the exact position of the nearest object pixel is determined for each image pixel as well. Further improvement was made by Leymarie and Levine [85], where the authors have proposed implementation which, in terms of computational complexity, is comparable to the chamfer (i.e., scalar) DT algorithms. Such an efficient implementation has been achieved by eliminating the floating-point operations such as multiplications and square root operations, as well as using only integer arithmetic. This reduces computational complexity of 4SSED algorithm to one equivalent to corresponding chessboard DT.

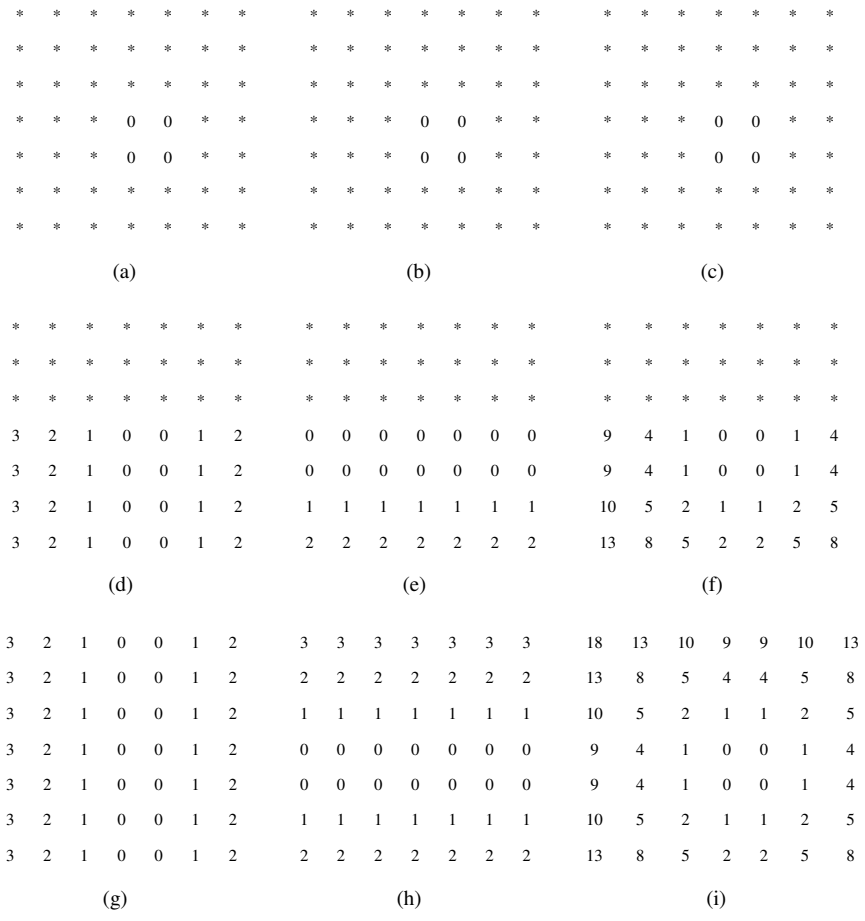


Figure 3.10: Illustration of implementation of 8SED algorithm to binary image in Fig. 3.6. The first row illustrates the original binary image after its initialization, where symbol $*$ denotes a suitably large integer. Second and third row illustrates the forward and backward super passes. In the first and second column two additional images are presented, consisting of the numbers of horizontal and vertical steps to the nearest object pixel, in a current super pass. The third column represents the sums of the squared integers contained in first two additional images in the same row.

4.2426	3.6056	3.1623	3.0000	3.0000	3.1623	3.6056
3.6056	2.8284	2.2361	2.0000	2.0000	2.2361	2.8284
3.1623	2.2361	1.4142	1.0000	1.0000	1.4142	2.2361
3.0000	2.0000	1.0000	0.0000	0.0000	1.0000	2.0000
3.0000	2.0000	1.0000	0.0000	0.0000	1.0000	2.0000
3.1623	2.2361	1.4142	1.0000	1.0000	1.4142	2.2361
3.6056	2.8284	2.2361	2.0000	2.0000	2.2361	2.8284

Figure 3.11: The final Euclidean distance map of the binary image, given in Fig. 3.6, using 8SED algorithm. The final Euclidean distances are computed using the square root of the distance image given in Fig. 3.10(i).

Finally, we want to mention something about the practical aspect of the described sequential EDT algorithms. For this purpose, two additional images of the same size as the original image are necessary, aimed to store the numbers of the vertical and horizontal steps of each image pixel to the nearest object pixel. The Euclidean distance assigned to each image pixel is then computed as the square root of the sum of these numbers squared. To illustrate the implementation of the sequential EDT algorithms, let us consider 8SED algorithm and its implementation to the binary image given in Fig. 3.6. Instead of comparing the square root of the sum of squared numbers of vertical and horizontal steps, we compare and store only the sum of the squared numbers. Thus avoiding operations with floating-point, we can accelerate algorithm implementation, as proposed in [85]. Complete implementation of such an algorithm is in Fig. 3.10, while the final Euclidean distance map, obtained by computing the square root of the distance image in Fig. 3.10(i), is shown in Fig. 3.11.

3.4 Three-dimensional distance transforms

In this section, we consider a digital space $\mathbb{Z}^3$ and distance $d : G \times G \to \mathbb{R}$ defined on the digital grid $G \subset \mathbb{Z}^3$, according to Definition 3.1. The Voronoi region associated with an arbitrary grid point $v \in \mathbb{Z}^3$ is now called a *voxel*, while a common boundary of two Voronoi regions is called a *Voronoi face*. Besides, a common boundary of two Voronoi faces is called *Voronoi edge*. Endpoints of the Voronoi edges are called the *Voronoi vertices* representing the common points of two Voronoi edges. Further, for each voxel there exist six voxels having a common face. These six voxels are also called *6-neighbors*, whereas the corresponding set of all 6-neighbors is called *6-neighborhood* of the voxel. In terms of the distance, the *6-neighborhood* consists

of all voxel neighbors at a city block distance 1. Also, twelve neighbors having a common edge are called the *edge neighbors* of the voxel, while the eight neighbors sharing the common vertex are called the *vertex neighbors* of the voxel. Finally, twenty six voxels having a common face, edge or vertex are called the *26-neighbors*, and set of all 26-neighbors is called a *26-neighborhood* of the voxel. In terms of distance, the *26-neighborhood* represents a set of all neighbors of the voxel located at a chessboard distance 1. Exploiting this, we have that the corresponding city block and the chessboard distances, defined on the digital grid $G \subset \mathbb{Z}^3$, is also called *6-neighbors* distance d_6, as well as the *26-neighbors* distance d_{26}, respectively. More formally, for any two integer lattice points $v = (v_x, v_y, v_z),\ w = (w_x, w_y, w_z) \in \mathbb{Z}^3$, the city block, chessboard and Euclidean digital distances in $\mathbb{Z}^3$ are defined as follows:

$$\begin{aligned} d_6(v,w) &= |v_x - w_x| + |v_y - w_y| + |v_z - w_z|, \\ d_{26}(v,w) &= max(|v_x - w_x|, |v_y - w_y|, |v_z - w_z|), \\ d_e(v,w) &= \sqrt{(v_x - w_x)^2 + (v_y - w_y)^2 + (v_z - w_z)^2}. \end{aligned} \tag{3.37}$$

Now we can talk about *path, local step, local distance* and *minimal (i.e., shortest) path* between two voxels. All these terms can be introduced in $\mathbb{Z}^3$ in a similar way as the corresponding terms in $\mathbb{Z}^2$, replacing the term pixel with term voxel. Thus a *path* $\mathcal{P}(v,w) = (v = v_0, v_1, ..., v_n = w)$ between two voxels $v, w \in \mathbb{Z}^3$ is defined as a sequence of voxels $v_i \in \mathbb{Z}^3$, where v_i and v_{i+1} are the neighboring voxels for all $i = 0, ..., n-1$. The ordered pairs of two neighboring voxels $(v_i, v_{i+1}),\ i = 0, ..., n-1$, are called the *local steps* along the path. The length of the path between two voxels is defined as a sum of the local distances assigned to each local step along the path, for a given metric. The path $\mathcal{P}(v,w)$ is the *shortest or minimal path* if it is of the minimal length among all the existing paths from v to w. The *distance between two voxels* is then defined as a *length of the minimal path between these two voxels*.

3.4.1 Chamfer distance transforms in 3D

As it has been mentioned, the computation of the three-dimensional DTs follows the same pattern as in the two-dimensional case. In the case of the chamfer DTs in 3D, the original binary image is first initialized in a way that the object voxels are set to zero, while the background voxels are set to infinity (i.e., a suitably large integer). Similar as in the 2D case, the chamfer DTs are computed in two passes over an image and can be completely described using two 3D distance masks, as presented in Fig. 3.12. In the *first pass*, the *forward mask F* is moved over an image from left to right, top to bottom, and front to back. In the *second pass*, the *backward mask B* is moved in opposite direction. The new distance value of the image voxel, covered by the center of the forward/backward mask, is the minimum of all the sums of the local distances

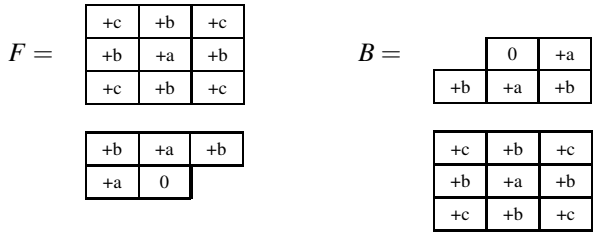


Figure 3.12: Masks for three-dimensional chamfer distance algorithms. Above submasks are at the upper level within each corresponding 3D mask.

and corresponding distance values assigned to image voxels covered by the mask.

The values of a, b and c within the corresponding distance masks, as given in Fig. 3.12, represent the local distances between the face, edge, and vertex neighbors, respectively. Depending on the choice of the values assigned to the local distances within the masks, the different 3D chamfer DT algorithms can be obtained. For example, for $a = 1,\ b = c = \infty$, or $a = b = c = 1$, we get 6-neighbors distance d_6, and 26-neighbors distance d_{26}, respectively. To evaluate the accuracy of these distances, Borgefors [18] estimated how much they differ from the exact Euclidean distance transform in 3D. An upper bound for 6-neighbors and 26-neighbors distance, for an image of size $N \times N \times N$, is equal to $-1.27N$ and $0.73N$, respectively. Exploiting this, it can be noticed that 6-neighbors (i.e., 26-neighbors) distance overestimates (i.e., underestimates) the corresponding exact Euclidean distances.

To provide a better approximation of the global Euclidean distance, the local distances a, b and c can be set to any real number. Thus in [18], the author considered the chamfer DT algorithms where the corresponding local distances are set to their true Euclidean lengths:

$$a = 1,\ b = \sqrt{2},\ c = \sqrt{3}. \tag{3.38}$$

Such a defined distance is called *quasi-Euclidean* distance in 3D, given it equals the exact Euclidean length of the shortest path between two voxels. A maximum of the difference from the Euclidean distance amounts $-0.147N$, making the quasi-Euclidean DT a very rough approximation of the exact EDT. In order to improve the accuracy of the approximation, Borgefors considered in [18, 22] the local distances which are set to their optimal values, where the term optimality refers to different criteria aimed to minimize the corresponding difference from the exact EDT. One such a possibility, introduced in [18], is given as follows:

$$a = 1,\ b \approx 1.31402,\ c \approx 1.62803, \tag{3.39}$$

where the corresponding upper bound of the difference from the EDT, for an image of the size of $N \times N \times N$, is approximately equal to $0.10N$.

As already mentioned in the 2D case, the use of real distance values in the corresponding DT algorithms can be the source of their weakness in terms of time complexity. Particularly, having in mind that all the necessary computations are carried out for three-dimensional distances. Consequently, it is of particular interest to use integer arithmetic rather than the corresponding real-valued one, since the integer local distances are more desirable from the aspect of their implementation regarding the time complexity. Hence, sub-optimal integer approximations of the local distances, obtained by multiplying the optimal real distances by corresponding integer factor and then rounding them to the nearest integers, can be of particular importance in such cases. In that context, starting from the above-mentioned optimal triplets of local distances (3.39) and scaling factor 3, the following integer distances are obtained:

$$a = 3,\ b = 4,\ c = 5 \tag{3.40}$$

where the upper bound of the difference from the exact EDT, for an image of the size of $N \times N \times N$, equals $-0.12N$. Again, we can conclude that the use of the appropriate sub-optimal local distances contributes to greater deviation from the exact global Euclidean distances, as already shown in (3.26).

3.4.2 Vector distance transforms in 3D

The two-dimensional sequential EDT algorithms, proposed by Danielsson [38], can easily be generalized to three dimensions. In this context, two sequential 3D EDT algorithms, called *6-neighbors sequential Euclidean distance* algorithm (6SED) and *26-neighbors sequential Euclidean distance* algorithm (26SED) have been proposed in [18] as 3*D* extensions of the already described 4SED and 8SED algorithms, respectively. As it can be expected, a larger number of the considered neighbors contributes to a closer approximation of the exact Euclidean distances as well. Contrary to the algorithms in 2D, herein the eight passes (arranged into the *forward* and *backward super passes*) over a volume image are necessary to propagating the relative coordinates to the nearest object voxel. At the end of the algorithms, each image voxel is assigned three integers representing the numbers of the local steps to the nearest object voxel along each image coordinate, and then the final distances are equal to the Euclidean length of the 3*D* integer vector assigned. Herein the original 3*D* binary image is first initialized using integer vectors $\mathbf{0} = (0,0,0)$ and $* = (*,*,*)$ (where $*$ again denotes a suitably large integer):

$$\mathbf{D}(x,y,z) \leftarrow \begin{cases} (0,0,0), & \text{if } (x,y,z) \text{ belongs to the object,} \\ (*,*,*), & \text{otherwise.} \end{cases} \tag{3.41}$$

As expected, both sequential 3D EDT algorithms can be described using the 3D distance-vector masks given in Fig. 3.13, where the masks describing 6SED algorithm are obtained by omitting the mask vectors $(1,1,1)$, $(0,1,1)$, $(1,0,1)$ and

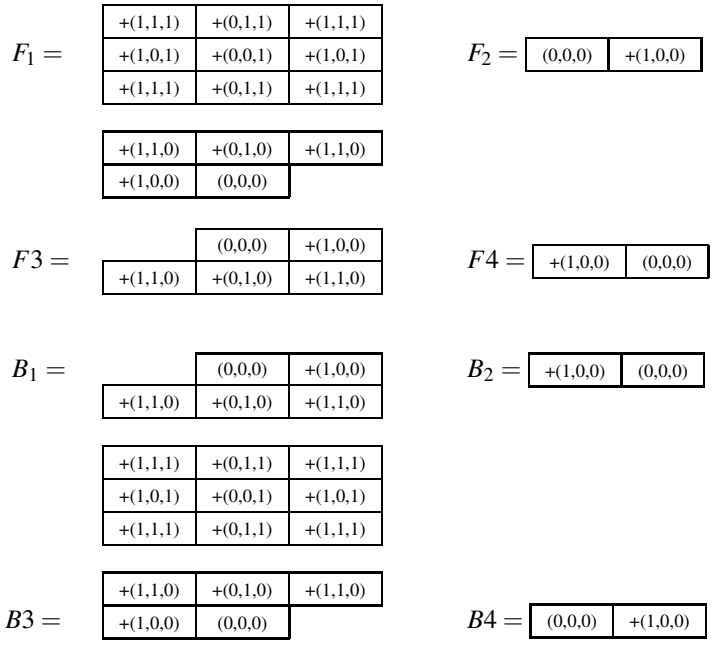


Figure 3.13: Masks for three-dimensional EDT algorithms. Above sub-masks are at the upper level within the masks corresponding.

$(1,1,0)$, given that only the face neighbors have been considered. In the *forward super pass*, the *forward mask* $F1$ is moved from the left to the right along the second row in the second image slice (plane), and then the mask $F2$ is moved back from the right to the left along the same row. These two passes are repeated for each row in the second slice. The third *forward mask* $F3$ is moved from the right to the left along the row before the bottom in the same slice, and then the mask $F4$ is moved back from the left to the right. These two passes are repeated for each row from the bottom to the top in the second image slice. All these four passes are repeated for each image slice in the volume image from the front to the back. In the *backward super pass*, the *backward masks* are moved similarly, but in the opposite direction. Based on this, the pseudo-code of both sequential 3D EDT algorithms is given in the following way:

Forward super pass :

$$
\begin{aligned}
&for\ z \leftarrow 2,\ldots,number\_of\_slices \\
&\quad for\ y \leftarrow 2,\ldots,number\_of\_rows \\
&\quad\quad for\ x \leftarrow 2,\ldots,number\_of\_columns\ //* \text{ the first pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ F_1} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k)) \\
&\quad\quad for\ x \leftarrow number\_of\_columns - 1,\ldots,1\ //* \text{ the second pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ F_2} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k)) \\
&\quad for\ y \leftarrow number\_of\_rows - 1,\ldots,1 \\
&\quad\quad for\ x \leftarrow number\_of\_columns - 1,\ldots,1\ //* \text{ the third pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ F_3} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k)) \\
&\quad\quad for\ x \leftarrow 2,\ldots,number\_of\_columns\ //* \text{ the fourth pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ F_4} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k))
\end{aligned}
$$

Backward super pass :

$$
\begin{aligned}
&for\ z \leftarrow number\_of\_slices - 1,\ldots,1 \\
&\quad for\ y \leftarrow number\_of\_rows - 1,\ldots,1 \\
&\quad\quad for\ x \leftarrow number\_of\_columns - 1,\ldots,1\ //* \text{ the fifth pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ B_1} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k)) \\
&\quad\quad for\ x \leftarrow 2,\ldots,number\_of\_columns\ //* \text{ the sixth pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ B_2} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k)) \\
&\quad for\ y \leftarrow 2,\ldots,number\_of\_rows \\
&\quad\quad for\ x \leftarrow 2,\ldots,number\_of\_columns\ //* \text{ the seventh pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ B_1} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k)) \\
&\quad\quad for\ x \leftarrow number\_of\_columns - 1,\ldots,1\ //* \text{ the eighth pass} \\
&\quad\quad\quad \mathbf{D}(x,y,z) \leftarrow \min_{(i,j,k)\ \in\ B_2} (\mathbf{D}(x+i,y+j,z+k) + \mathbf{v}(i,j,k))
\end{aligned}
\tag{3.42}
$$

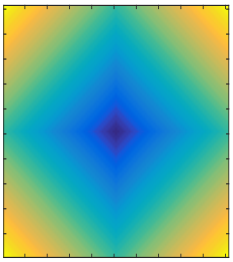
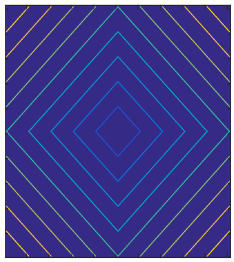
where $\mathbf{D}(x,y,z)$ represents the distance vector assigned to image voxel at (x,y,z) position, $\mathbf{v}(i,j,k)$ is a local distance vector at (i,j,k) position in the vector mask (cen-

tered at $(0,0,0)$), and number_of_slices, number_of_rows and number_of_columns are the numbers of image slices, rows and columns, respectively. Again it can be noticed that, for the sake of simplicity, in the pseudo-code presented we have used the notation $\mathbf{v} \leftarrow \min_{\omega \in V}(\omega \big| \omega \in V)$, instead of $\mathbf{v} \leftarrow \arg\min_{\omega \in V} \big|\omega\big|$.

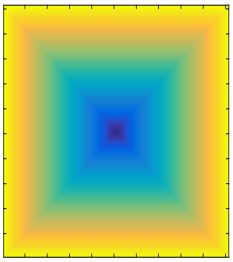
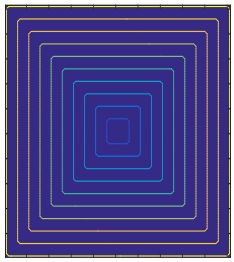
3.5 Illustrations of DTs in 2D and 3D

This section illustrates the properties of DT algorithms in 2D and 3D that have already been considered and described in this chapter. In the case of 2D algorithms, we observe the performances related to the visual characteristics of the associated distance maps, the shape of propagation of the assigned distances, and their local distribution over an image. In that context, Fig. 3.14, from the top to the bottom, presents the distances computed from the central pixel (in an image of a size of 100×100 pixels) using d_4, d_8, chamfer $d_{(3,4)}$ (given in (3.26)), and 8SED propagation-based DT algorithm, respectively. In the first column the distance maps are shown, generated by corresponding DT algorithms, where lighter color corresponds to a greater assigned distance value. To provide a better illustration of the performance of the algorithms considered, we observe the propagation of the computed distances in the corresponding distance maps. The second column of Fig. 3.14 presents digital circles (associated with appropriate DT) propagated from the central pixel over the whole image. It can be noticed that the d_4 and d_8 metrics provide a rough approximation of the circular (i.e., Euclidean) distance propagation. Thus, d_4 metric generates a *diamond-shape distance propagation* where the observed digital circles are squares rotated by an angle of $\pm 45^\circ$, whereas d_8 metric produces a *square-shape distance propagation* with squares as the corresponding digital circles. On the other hand, the chamfer DT algorithm $d_{(3,4)}$ propagates the octagons over an image, which as such represent a better approximation of the appropriate exact Euclidean circles. For 8SED algorithm, we observe that the distance propagation is closest to the exact circular (i.e., Euclidean) distance propagation, whereas the observed circles are then the exact Euclidean digital circles. To provide an additional illustration, we present, in the third column of the figure, the central 7×7 sub-maps of the appropriate distance maps, which serve to illustrate how the central distances are distributed over the corresponding distance map. Odd distances are underlined for clarity. It should be noted that in order to obtain the exact distances assigned to image pixels, it is necessary to divide the values presented in the third row by 3. On the other hand, regarding the 8SED algorithm, the generated values which are displayed in the fourth row represent the squares of the corresponding Euclidean distances. To obtain final Euclidean distances, it is necessary to compute the square root of the values assigned to each image pixel.

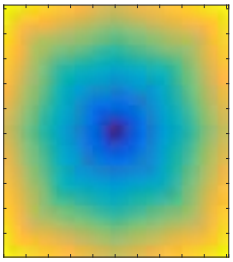
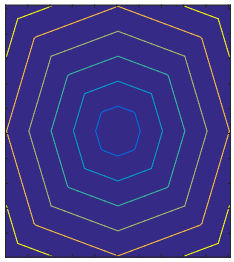
Finally, it should be noted that the algorithms presented can be of particular importance in a variety of image processing and computer vision tasks and applications.

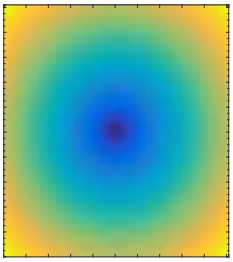
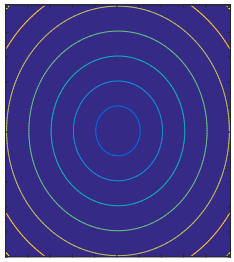
6	5	4	3	4	5	6
5	4	3	2	3	4	5
4	3	2	1	2	3	4
3	2	1	0	1	2	3
4	3	2	1	2	3	4
5	4	3	2	3	4	5
6	5	4	3	4	5	6

3	3	3	3	3	3	3
3	2	2	2	2	2	3
3	2	1	1	1	2	3
3	2	1	0	1	2	3
3	2	1	1	1	2	3
3	2	2	2	2	2	3
3	3	3	3	3	3	3

12	11	10	9	10	11	12
11	8	7	6	7	8	11
10	7	4	3	4	7	10
9	6	3	0	3	6	9
10	7	4	3	4	7	10
11	8	7	6	7	8	11
12	11	10	9	10	11	12

18	13	10	9	10	13	18
13	8	5	4	5	8	13
10	5	2	1	2	5	10
9	4	1	0	1	4	9
10	5	2	1	2	5	10
13	8	5	4	5	8	13
18	13	10	9	10	13	18

Figure 3.14: First column, from the top to the bottom, presents distance maps consisting of the distances from the single central pixel utilizing d_4, d_8, $d_{(3,4)}$ (3.26), and 8SED propagation based DT algorithms; the second column illustrates the corresponding digital circles propagated over an image starting from the central pixel; and the third column contains the central 7×7 sub-maps of the corresponding distance maps, aimed to illustrate the distribution of the assigned distances in the corresponding distance maps. Odd distances are underlined for clarity.

We will provide more about this in the following section, while herein we emphasize only several important aspects of their applications that relate to certain requirements and restrictions. In this sense, if the essential requirement is to compute the distances as close as possible to the exact Euclidean distances, then 8SED algorithm [38] is of particular importance. However, as it has already been mentioned, this algorithm requires real-valued arithmetic, as well as two additional images of the same size as the original one to store the intermediate results of the algorithm. Thanks to the improvements proposed in [85], implementation of 8SED can be further improved and accelerated as well. On the other hand, the simplest, but also the fastest among all DT algorithms presented in this chapter, is d_4 DT algorithm, providing at the same time the roughest approximation of the exact EDT. Such a property comes from its simple spatial complexity which contributes to its poor accuracy. Taking into account these considerations, we have that if the speed of the algorithm is more important than its accuracy, then d_4 metric can serve to obtain a preliminary insight into the distance distribution over an image, as shown in Fig. 3.14. Otherwise, in all other applications and tasks, its applicability and usefulness is rather limited and focused only on its theoretical aspects.

On the other hand, to illustrate the performance of several DT algorithms in 3D, we will follow the concept similar to 2D algorithms where, due to limitations regarding the visualization of the distance maps generated, we observe only the shape of the digital spheres generated using the corresponding DT algorithm. The obtained digital spheres of radius of 20 voxels, centered within the central voxel (of an image of $100 \times 100 \times 100$ voxels) using d_6, d_{26}, $d_{(3,4,5)}$ (given in (3.40)), and 26SED algorithms are shown in Fig. 3.15. It can be noticed that a coarse approximation of the exact Euclidean sphere is clearly visible for d_6 and d_{26} DT algorithms, where the associated digital spheres are the *octahedron* (Fig. 3.15(a)) and *cube* (Fig. 3.15(b)), respectively. On the other hand, as it has already been noticed, a better approximation is provided using $d_{(3,4,5)}$ DT algorithm where the associated digital sphere is *deltoidal icositetrahedron* consisting of 24 quadrilateral faces, whereas, in the case of 26SED propagation-based algorithm, the corresponding digital sphere is actually exact Euclidean digital sphere. Such a property of 26SED algorithm makes it as the most accurate propagation-based DT algorithm among the other 3D algorithms already discussed in this chapter. Following this, if it is necessary to achieve as close as possible approximation of the exact Euclidean distances, then the 26SED algorithm is rather a recommending one. However, its drawbacks are related to the execution time and memory requirements, given that three additional images of the same size as the original image are necessary to store three integers assigned to each image voxel. These three integers represent the numbers of steps to the nearest object voxel in all three image directions. The final Euclidean distance assigned to image voxel is then equal to the square root of the squares of these three stored integers. Similarly, as for 2D, the 26SED algorithm can be significantly simplified and accelerated if integer

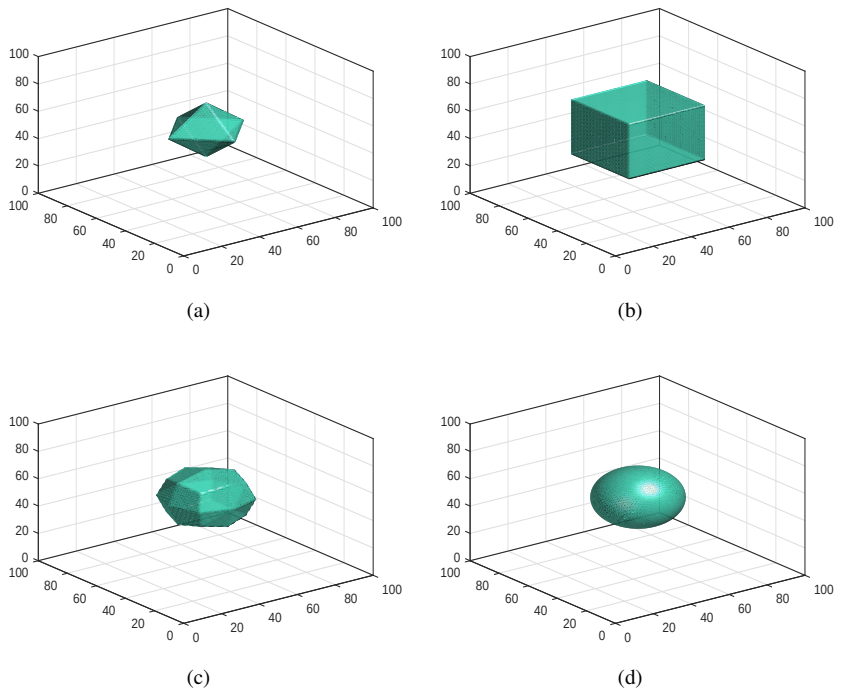
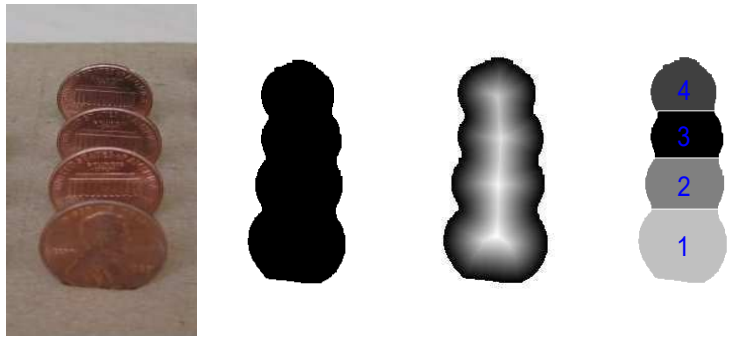


Figure 3.15: Illustration of the digital spheres with radius of 20 voxels, centered in the central image voxel, computed using the d_6 (a), d_{26} (b), $d_{(3,4,5)}$ (c) and 26SED (d) propagation based algorithms.

arithmetic is applied, instead of the real-valued arithmetic. This can be achieved by extending the concept, originally presented in [85] for 2D, where, instead of applying square root operations and manipulation with intermediate floating-point results in each algorithm step, we are working only with integers representing the sum of the squares of the stored integers. The final Euclidean distances are then computed as the square root of such integers. It should be noted that this algorithm will be of our particular interest in research presented in the following chapter, where we will show that performance in terms of increased precision and accuracy, together with increased invariance with respect to rotation and translation, can be significantly improved if the coverage information about voxel, available in the corresponding voxel coverage image representation, is treated appropriately. In that context, we will present two novel EDT algorithms that are theoretically well-designed and experimentally evaluated in the following chapter.



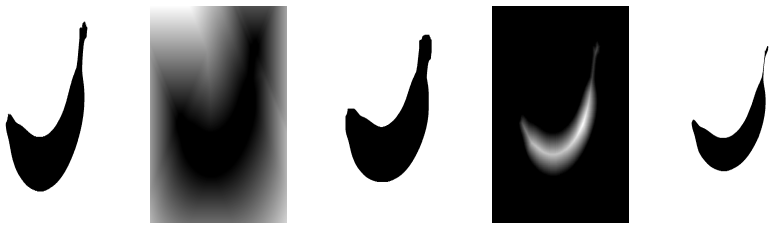
(a) (b) (c) (d)

Figure 3.16: Illustrations of DT application in counting the imaged overlapping coins. (a) Original image; (b) the corresponding binary image with only one connected component; (c) the distance field of the complement of (b); and (d) watershed segmentation applied to the negative of the image in (c), where each number denotes a single connected component.

3.6 Applications of DTs

As we have already mentioned, in this section, we will briefly present several applications of DT in various image processing and computer vision tasks. Given that the concept of DT is one of the fundamental, but also a generic tool in various shape-based analysis tasks, we present herein several experiments to illustrate importance and usefulness of DT algorithms, and also to provide a better understanding of the theoretical considerations discussed in this chapter.

- *Separation of overlapping objects using the watershed image segmentation* [158]: Figure 3.16 illustrates the applicability of DT in the task of counting the image objects (i.e., coins) shown in Fig. 3.16(a). The most common challenge appearing in such tasks is a possibility of the existence of overlapping objects that may be presented in the appropriate binary image as one connected component (see Fig. 3.16(b)). Considering this, the corresponding separation of such a connected component is one of the most essential steps in counting the image objects observing the number of the connected components in the binary image observed. This can be achieved by computing DT to the complement of the binary version of original image aimed to obtain the corresponding distance map (given in Fig. 3.16(c)), and then perform the watershed segmentation to the negative distance map. Following this, the overlapping objects are then correctly separated, and each object is then presented as a single connected



(a) (b) (c) (d) (e)

Figure 3.17: The computation of dilation and erosion of a binary image using DT. (a) Original binary image; (b) distance map of image in (a); (c) thresholded version of (b) using the Otsu thresholding method [107]; (d) distance map of the complement of (a); and (e) thresholded version of (d). Images in (c) and (e) are the same as those obtained by dilation and erosion using a structuring ball of a size equal to threshold applied to (b) and (d), respectively.

component (see Fig. 3.16(d)). In such a way, it is established a one-to-one correspondence between the number of image objects and their corresponding associated components, what was our initial intention.

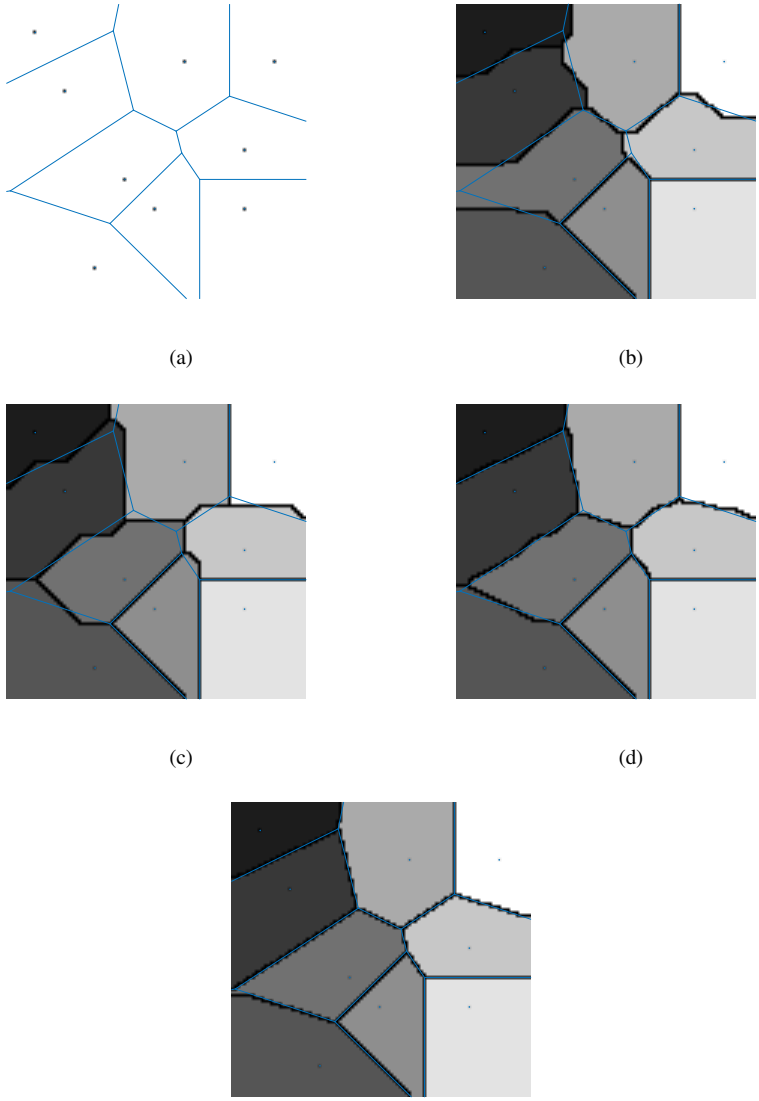
- *The computation of morphological operations such as dilation, erosion, opening, closing, etc.* [35, 111]; For an illustration of this type of DT application, we consider dilation and erosion of binary image (given in Fig. 3.17(a)) where the structuring element is a ball of a given metric. Assume that the original image is first transformed into the distance map (Fig. 3.17(b)), and let suppose that such an obtained distance map is then thresholded using a threshold equal to the radius of a given structuring element. The resulting image (given in Fig. 3.17(c)) is then the same as a dilated original image using the same structuring element. Similarly, if the corresponding DT is applied to the complement of the original image (Fig. 3.17(d)), and then thresholded at a level defining the structuring element, the resulting image is then the same as the eroded original image using the same structuring ball (Fig. 3.17(e)). Regarding the other morphological operations, the applicability of DTs follows from the fact that morphological operations can be expressed as a combination of dilation, erosion and corresponding set operations, whose connection with DT algorithms has been already discussed.

- *Computing the geometrical representations and shape descriptors* such as *skeletonization, Voronoi diagrams, medial axis transforms, etc.* [23, 32, 33, 130, 155]; To illustrate applicability of DT algorithms in such defined image processing tasks, we have considered a Voronoi tessellation of a plane generated by a set of 9 influence points given as shown in Fig. 3.18(a). For an illustration, we observe d_4, d_8 and quasi-Euclidean $d_{(1,\sqrt{2})}$ (3.20) DT algorithm, along with parallel EDT algorithm proposed in [174]. The four obtained Voronoi tessellations associated with the corresponding DT algorithms are shown in Fig. 3.18. Such an obtained Voronoi tessellations are also referred to as *pseudo-Voronoi tessellations*, due to their discrete (i.e., digital) nature, as well as dependence on the choice of the corresponding DT algorithm. The exact Euclidean tessellation is also included to illustrate the accuracy of the generated (pseudo-)Voronoi tessellations. In that sense, each Voronoi region is labeled by a certain grey level, whereas the Voronoi diagram generated using the exact Euclidean distance is given in color. As it can be seen, the resulting pseudo-Voronoi tessellations can deviate significantly depending on the metric that is used. Following this, most image pixels assigned to wrong Voronoi regions are visible in Fig. 3.18(b) and 3.18(c), in the case of d_4 and d_8 DT algorithms. Large portions of such pixels are particularly visible and present close to the left edge of the image in the middle, as well as around the center of the image. As expected, in the case of quasi-Euclidean DT algorithm $d_{(1,\sqrt{2})}$ (3.20) (given in Fig. 3.18(d)), the generated results are improved in terms of achieved accuracy and precision, even though several pixels being still assigned to wrong Voronoi regions. Finally, the parallel EDT algorithm [174] provides the exact digital Voronoi tessellation based on the Euclidean distance, as shown in Fig. 3.18(e).

- *Robot navigation* to determining the shortest path from one place to another among the various obstacles [29, 36, 59, 138]; One such example, originally presented in [68], is illustrated in Fig. 3.19(a) where two given points, marked as blue and red stars, are considered as the *starting* and *destination point* of the motion of the robot, respectively, whereas the points given in white represent obstacles (e.g., walls). For an illustration, we have considered a quasi-Euclidean DT algorithm $d_{1,\sqrt{2}}$ (3.20). In the first step, distances are propagated over an image from the starting point as the source of propagation. Figure 3.19(b) illustrates such a generated distance map where each background pixel is assigned its shortest distance to the starting point. Similarly, in the second step, the distances are propagated over an image, but now from the destination point as a source of propagation. Such an obtained distance map is shown in Fig. 3.19(c). It can be noticed that both generated distance maps are illustrated as grey-level images where the brighter pixels correspond to larger assigned distances. If these two maps are summed then the pixels, to which the minimal values are assigned, belong to the shortest paths. Such a generated grey-level

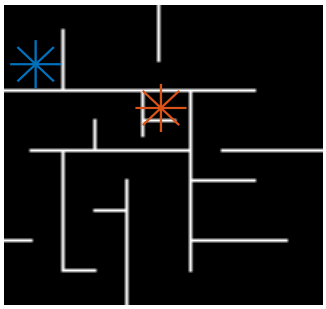
distance map is given in Fig. 3.19(d), where the pixels belonging to the shortest paths are shown darker. As it can be noticed, we have generated a number of different shortest paths connecting the starting and destination points, and among them, it is necessary to pick only one. There exist many different ways to achieve this, and herein we have decided for a morphological thinning as one possibility to obtain only one shortest path between two given points (shown as a white path in Fig. 3.19(e)).

- Other DT applications in shape-based image analysis tasks are: *shape matching* [19, 20, 108], *shape measures (descriptors) based on the distance* [36, 117, 118], where the distribution of the distances in the corresponding distance map can be used as a shape descriptor, *image registration* [36, 80], etc. More about application of DT algorithms can be found in [21, 53].



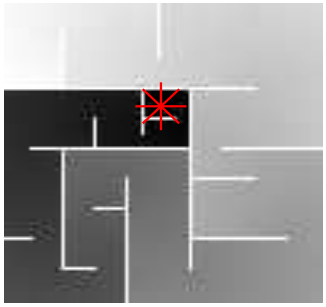
(a) (b) (c) (d) (e)

Figure 3.18: (a) A Voronoi tesselation of a plane generated with 9 influence points. Examples of (pseudo-)Voronoi tessellations for the same influence points using the different DT algorithms: (b) city block, (c) chessboard, (d) quasi-Euclidean and (e) EDT algorithm [174].

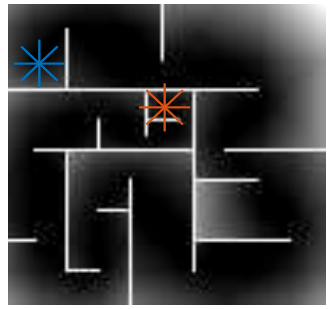


(a)

(b)



(c)



(d)

(e)

Figure 3.19: Example of the shortest path computation among the obstacles using the quasi-Eucliden DT (3.20). The starting and destination points are marked as blue and red stars, respectively. In (b), (c) and (d), the lighter pixels, the larger assigned distances. The darker pixels in (d) belong to one or more of the shortest paths between the source points. Path in white in (e) is picked as particular unique shortest path using the morphological thinning.

Chapter 4

Sub-voxel precise Euclidean distance transforms in 3D

This chapter illustrates a relatively novel approach to improving the performance of 3D EDT algorithms based on information contained in the voxel coverage image representation. Before that, let us recall that DTs are usually defined on a binary image as a mapping which assigns each background voxel the distance from its center to the center of the nearest object voxel. However, such an image representation can be closely related to possible irreversible loss of important information about the geometry of the imaged object. This loss of the object information, particularly on the object boundary, can further lead to negative effects to computed EDTs in terms of decreased precision and accuracy, as well as reduced invariance to rotation and translation.

One possibility to deal with these challenges is to utilize the coverage image representation defined in Chapter 2, instead of the binary image representation itself. As it has been already mentioned, in this representation, each image voxel is assigned value proportional to its relative volume covered by the observed imaged object. Such assigned values are then used to estimate the position of the object within the boundary voxel. Advantages of utilizing the coverage image representation, instead of the corresponding classical binary one in the tasks of improving the accuracy of EDT are presented in [63] for the 2D case. This result has motivated further research on how the coverage model can be utilized to develop EDTs with sub-element precision. The initial studies on utilizing the coverage model to achieve more accurate 3D EDTs with sub-voxel precision have been presented in [91]. Since then this study has motivated further research to develop other sub-voxel precise EDT algorithms based on the coverage model. In this chapter, we illustrate how considerable improvement in the performance of Euclidean DTs can be achieved if the voxel coverage information

is used to estimate the sub-voxel position of the object within the voxel. Such an obtained precise estimate of the boundary position can be further utilized to obtain more precise and more accurate 3D EDTs. In the following, we present two novel methods of linear complexity, originally proposed in [71], for computing the EDT estimation with sub-voxel accuracy, which can then be used to improve performance of any vector propagation DT algorithm. The statistical evaluation of the proposed methods by their comparison both with the classical vector propagation-based EDT using 26-neighbors [18], and with the already existing method utilizing the voxel coverage representation and sub-voxel position of the object boundary [91]. The evaluation results clearly illustrate the performance improvement of our proposed methods in terms of the decreased bias and variance, as well as reduced variance to rotation and translation, compared to the other two competitive 3D EDT algorithms.

4.1 Sub-voxel estimate of the boundary position using voxel coverage information

As it has already already said in this chapter, we will present new methods for computing EDTs that utilize voxel coverage information to estimate the position of the object boundary within the voxel. Herein we show how such a sub-voxel position of the object boundary can be further used to improve the performance of any vector propagation-based DT algorithm.

In that sense, let first suppose that the boundary of the object covering the voxel is locally planar (as shown in Fig. 4.1a). Given this, it is easy to provide an estimate of the object boundary position within the voxel using its coverage if the normal direction of the boundary is known. In that context, let us notice that there exists a functional relationship between the signed distance from the object boundary to the center of the voxel, herein denoted with d_v, the volume voxel coverage v, as well as the normal direction of the planar object boundary $\mathbf{n}$. Before we start with the derivation of the main result of the paper [71], let first note that, for all normal directions $\mathbf{n}$, relation $v = 0.5$ is equivalent with $d_v = 0$, while for each normal direction parallel to one of the voxel coordinate axes, the linear relation $d_v = 0.5 - v$ holds true. Otherwise, in the case where the object normal direction is not parallel to the coordinate axes, this linear relation can be only used as a simple approximation of the signed sub-voxel distance d_v if a normal direction of the object boundary is not known. Such a linear approximation has been first proposed by Linnér and Strand in [91], and in the following, it will be denoted as d_v^{lin}:

$$d_v^{lin} = 0.5 - v. \tag{4.1}$$

However, in the case when the normal direction of the object boundary is known, the exact sub-voxel position of the object boundary within the voxel can be expressed

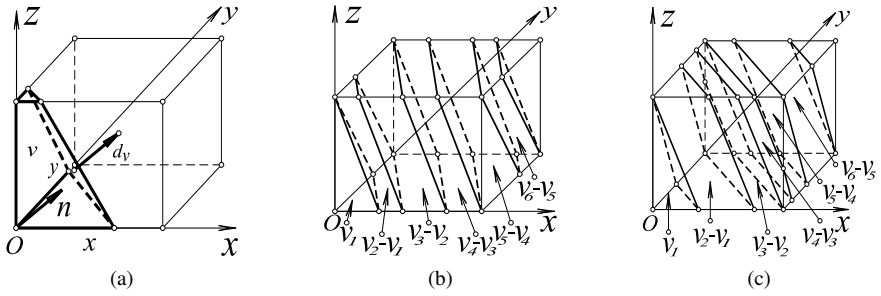


Figure 4.1: (a) The normal direction of the object boundary $\mathbf{n}$, signed distance d_v, volume coverage v (bolded volume), and distance x representing the intersection point of the object boundary and x-axis. (b) Volumes v_1, v_2, v_3, v_4, v_5 and v_6 when $n_y + n_z \leq n_x$ and (c) when $n_y + n_z > n_x$.

accurately. Indeed, the position of a planar object boundary within the voxel is uniquely defined by its unit normal direction $\mathbf{n} = (n_x, n_y, n_z)$, as well as the signed sub-voxel distance d_v from the voxel centre (where we choose orientation such that $v < 0.5 \Rightarrow d_v(v) > 0$). For this purpose, we present an algorithm for computing the exact distance d_v (i.e., the exact object boundary position within the voxel) as a function of the voxel coverage v and the normal direction of the object boundary $\mathbf{n}$. This algorithm also represents the main (i.e., original) contribution of the work [71], and herein we present a complete theoretical background that enables that this algorithm will be theoretically well-founded, and its derivation mathematically correct.

In addition, it can be noticed that, for any normal direction $\mathbf{n}$, the signed distance $d_v(v)$ is antisymmetric function around $v = 0.5$, i.e., it holds that $d_v(1-v) = -d_v(v)$ and $d_v(0.5) = 0$ as well. Following this, it is enough to observe only the volume coverages ranging through the interval $[0,\ 0.5]$, as well as the normal directions $\mathbf{n} = (n_x, n_y, n_z)$ for which $n_x \geq n_y \geq n_z \geq 0$. Besides, it is easy to notice that the other cases are based on symmetry, and can be derived by changing the sign or swapping the place of n_x, n_y and n_z. In order to make the derivation of the main result easier, let introduce the Cartesian coordinate system in a way as in Fig. 4.1(a), and let consider as well the plane with the normal direction $\mathbf{n} = (n_x, n_y, n_z)$ which passes through the origin (i.e., voxel vertex) $O(0,0,0)$. By translating such a plane from the vertex $(0,0,0)$ to the vertex $(1,1,1)$, we obtained the volumes v_1, v_2, v_3, v_4, v_5 and v_6 (as presented in Fig. 4.1) representing the parts of the voxel volume to the left of the plane when it passes the corresponding voxel vertex. The order in which the plane passes through voxel vertices depends on the relation between n_x and $n_y + n_z$. For example, in the case of $n_y + n_z \leq n_x$ the order of vertices is $(0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1)$ and $(1,1,0)$ (Fig. 4.1(b)), whereas in the case of $n_y + n_z > n_x$ this order becomes

$(0,0,1),(0,1,0),(1,0,0),(0,1,1),(1,0,1)$ and $(1,1,0)$ (Fig. 4.1(c)). Using an elementary geometry, we can express these volumes for a given normal direction $\mathbf{n} = (n_x, n_y, n_z)$, as follows:

$$
\begin{aligned}
& v_1 = \tfrac{1}{6}\tfrac{n_z^2}{n_x n_y}, \qquad v_2 = \tfrac{1}{6}\tfrac{n_z^2 - 3n_y n_z + 3n_y^2}{n_x n_y}, \\
& \text{if } (n_y + n_z \le n_x) \\
& \qquad v_3 = \tfrac{1}{2}\tfrac{n_y + n_z}{n_x}, \\
& \text{else} \\
& \qquad v_3 = \tfrac{1}{6}\tfrac{n_z^2 - 3n_x n_z + 3n_x^2}{n_x n_y} - \tfrac{1}{6}\tfrac{(n_x - n_y)^3}{n_x n_y n_z}, \\
& v_4 = 1 - v_3, \qquad v_5 = 1 - v_2, \qquad v_6 = 1 - v_1.
\end{aligned}
\tag{4.2}
$$

Now, thanks to these observations and the results derived, we are able to present the main contribution of the paper [71], i.e., to present the complete algorithm for computing the signed distance d_v as a function of the volume coverage v, $0 \le v \le 0.5$, for a given normal direction $\mathbf{n} = (n_x, n_y, n_z)$:

Algorithm 1

Input: The normal direction of the object boundary $\mathbf{n} = (n_x, n_y, n_z)$ *with* $n_x \ge n_y \ge n_z$, *and voxel coverage* v *where* $0 \le v \le 0.5$.
Output: The signed distance d_v *from the voxel centre to the object boundary.*

---

if $(n_y = 0)$ /* *implies* $n_z = 0$ */
  $d_v = 0.5 - v$
elseif $(v \le v_1)$
  $d_v = \frac{1}{2}(n_x + n_y + n_z) - \sqrt[3]{6 v n_x n_y n_z}$
elseif $(v \le v_2)$
  $d_v = \frac{1}{2}(n_x + n_y) - \sqrt{2 v n_x n_y - \frac{1}{12} n_z^2}$
elseif $(v \le v_3$ and $n_z \ne 0)$
  $a = \frac{n_z^3}{n_y^3}$; $b = -3\frac{(n_y + n_z) n_z^3}{n_x n_y^3}$
  $c = 3\frac{n_z^3 (n_y^2 + n_z^2)}{n_x^2 n_y^3}$; $d = 6v\frac{n_z^4}{n_x^2 n_y^2} - \frac{n_z^3 (n_y^3 + n_z^3)}{n_x^3 n_y^3}$
  $x = CardanRoots(a, b, c, d)$
  if $(ny + nz \le nx)$
    select $x \in [\frac{n_y}{n_x}, \frac{n_y + n_z}{n_x}]$
  else

select $x \in [\frac{n_y}{n_x}, 1]$
endif
$d_v = \frac{1}{2}(n_x + n_y + n_z) - xn_x$
else
if $(ny + nz \leq nx)$
$x = v + \frac{1}{2}\frac{n_y+n_z}{n_x}$
else
$a = 2\frac{n_z^3}{n_y^3}$; $b = -3\frac{n_z^3}{n_y^3}\frac{n_x+n_y+n_z}{n_x}$
$c = 3\frac{n_z^3}{n_y^3}\frac{n_x^2+n_y^2+n_z^2}{n_x^2}$; $d = 6v\frac{n_z^4}{n_x^2 n_y^2} - \frac{n_z^3}{n_y^3}\frac{n_x^3+n_y^3+n_z^3}{n_x^3}$
$x = CardanRoots(a,b,c,d)$
select $x \in [1, \frac{n_y+n_z}{n_x}]$
endif
$d_v = \frac{1}{2}(n_x + n_y + n_z) - xn_x$
endif

To provide a better understanding of the presented algorithm, it should be noted that in some sub-cases it is necessary to make the algorithm capable to compute the roots of the third-degree polynomial. Herein we have decided to implement Cardano formula, the function $CardanRoots(a,b,c,d)$, which solves the cubic equation $ax^3 + bx^2 + cx + d = 0$ for $a,b,c,d \in \mathbb{R}$. More about the Cardano formula itself can be found in *Appendix B* given at the end of thesis. Taking into account results related to Cardano formula (16), the roots of this cubic equation can be expressed as follows:

$$\begin{aligned} x_1 &= S + T - \tfrac{b}{3a}, \\ x_2 &= -\tfrac{S+T}{2} - \tfrac{b}{3a} + \tfrac{i\sqrt{3}}{2}(S - T), \\ x_3 &= -\tfrac{S+T}{2} - \tfrac{b}{3a} - \tfrac{i\sqrt{3}}{2}(S - T), \end{aligned}$$

where

$$S = \sqrt[3]{R + \sqrt{Q^3 + R^2}}, \qquad T = \sqrt[3]{R - \sqrt{Q^3 + R^2}},$$

and

$$Q = \frac{3ac - b^2}{9a^2}, \qquad R = \frac{9abc - 27a^2d - 2b^3}{54a^3}.$$

In order to illustrate the behaviour of d_v as a function of the volume coverage v, the plots of function $d_v(v),\ 0 \leq v \leq 1$, for three given normal directions $\mathbf{n} = (n_x, n_y, n_z)$,

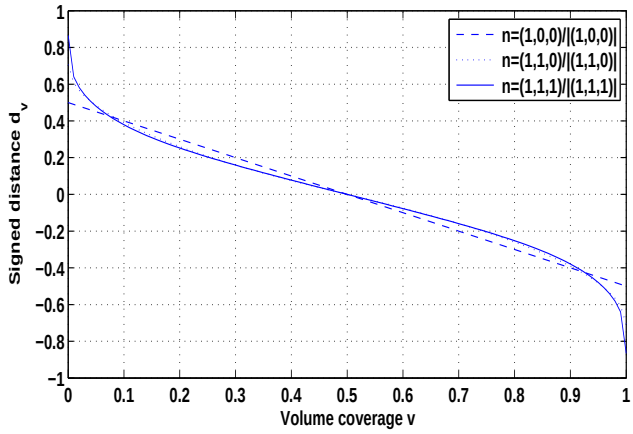


Figure 4.2: Signed distance function $d_v(v), 0 \leq v \leq 1$, for three different unit normal directions $\mathbf{n} = (n_x, n_y, n_z)$.

are shown in Fig. 4.2. It can be noticed that, for a fixed normal direction $\mathbf{n}$, the function $d_v(v), 0 \leq v \leq 1$ is continuous and monotonically decreasing, as well as $d_v(1-v) = -d_v(v)$. Besides, we can notice that d_v can be very well approximated using a linear function $d_v^{lin} = 0.5 - v$, which is obtained from the Algorithm 1 for normal direction $\mathbf{n} = (1,0,0)$. Better visualization of the signed distance function d_v is shown in Fig. 4.3, where d_v is plotted as a function of the normal direction $\mathbf{n}$ for 6 given voxel coverage values. It can be noticed that the top surface, for $v = 0$, peaks at $\sqrt{3}/2$ and for $\mathbf{n} = (1,1,1)$, while the bottom surface, relating to $v = 0.5$, is constant equal to zero, and it is equivalent with $d_v(0.5) = 0$ for all normal directions $\mathbf{n}$. Also, it can be noticed that the function plotted is both smooth and bounded, and, as such, it can be tabulated and interpolated with pretty good accuracy. Taking these considerations into account, the certain rather tedious and demanding computations of d_v can be avoided, which could significantly accelerate the execution of the algorithm itself in the sequential EDT implementation.

4.2 Proposed 3D Euclidean distance transforms

In this section, we present how the signed sub-voxel distance d_v, obtained as the output of the Algorithm 1, can be utilized to improve the estimation of the Euclidean distances of any vector propagation-based 3D EDT algorithm. Based on this, we will be able to propose novel methods for estimating the Euclidean distance transforms, with sub-voxel accuracy. Of particular importance in our derivation is a clas-

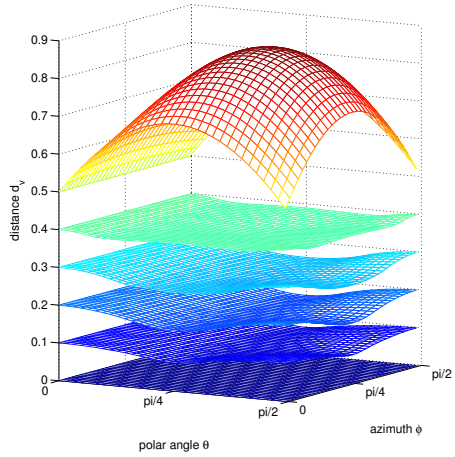


Figure 4.3: The signed distance function $d_v(\mathbf{n})$ for a given voxel coverage values $v \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$ (from the top to the bottom) and normal directions in the first octant, parametrized by spherical coordinate angles Θ and Φ

sical sequential vector propagation-based EDT algorithm [18] that has been already described in Chapter 3. Such a defined propagation-based EDT algorithm, herein referred to as binary EDT (BEDT), computes for each background voxel a distance vector $\mathbf{d}_i = (x_i, y_i, z_i)$ pointing to the closest object voxel, and assigns to that background voxel the distance value computed as:

$$d_i = d_{BEDT} = \sqrt{x_i^2 + y_i^2 + z_i^2}. \tag{4.3}$$

To provide an improvement of accuracy of BEDT algorithm, the authors in [91] proposed a linear approximation of d_v, as given in (4.1), to define Anti-Aliased Euclidean Distance Transform (AAEDT) as follows:

$$d_{AAEDT} = d_{VCEDT1} = d_{BEDT} + d_v^{lin} = d_{BEDT} + 0.5 - v. \tag{4.4}$$

To improve the performances of such a proposed EDT approximation, the authors of the [91] showed that accuracy can be further improved by using other non-standard discretization grids such as Body-Centered Cubic (BCC) and Face-Centered Cubic (FCC). However, in our research, we have focused on improving the performance of the EDT estimates by increasing accuracy of the boundary position within the voxel utilizing only the information available from its coverage representation on the Cartesian Cubic grid. One possibility to achieve this improvement is based on the appropriate treatment of information about the normal direction of the object boundary

within the voxel. Such an approach, originally proposed in [71], will be exploited in the following to improve the performance of the 3D EDT estimation.

To involve the information about the normal direction of the object boundary into the computation of the signed distance d_v, we have set one simple, but also a reasonable assumption. Let us assume that the boundary of the object is approximately orthogonal to the distance vector $\mathbf{d}_i$ assigned to the background voxel. Such an established assumption is asymptotically true in the sense that for the voxels close to the object boundary it may be violated, while it becomes asymptotically more exact for the voxels which are more and more distant from the object boundary. As a result, this assumption provides a sufficiently good approximation of the boundary position for the voxels which are further away from the object boundary. Thanks to this assumption, we can compute the signed distance d_v as the output of the Algorithm 1. Such an estimation of d_v, herein denoted with d_v^{ort}, can be used to assign the distance value to each background voxel computed as follows:

$$d_{VCEDT2} = d_{BEDT} + d_v^{ort}. \tag{4.5}$$

However, as already mentioned, this assumption may be not satisfied with the voxels close to the boundary of the object. To improve the sub-voxel EDT estimation for such voxels and their assigned distances, it is necessary to treat them with special care to enable additional information relevant to our task. For example, one such an approach is based on the use of local gradient vector assigned to boundary voxel to estimate the normal direction $\mathbf{n}$. This reasonable assumption can further lead to increased accuracy of the boundary position estimate within such voxels. For this purpose, we have decided to estimate the normal direction in each boundary voxel utilizing three $3 \times 3 \times 3$ Zucker-Hummel gradient filters [184]. In that context, the signed distance d_v, computed as the output of the Algorithm 1 using gradient-based normal direction, herein denoted with d_v^{grad}, can be used to improve EDT estimate for these voxels, instead of d_v^{ort}. Regarding the voxels located further away from the object boundary, we follow the same approach as in d_{VCEDT2}. Thus, if d_a and d_b denote the distances defined as:

$$d_a = \mathbf{d_i} \cdot \mathbf{n}, \quad d_b = |\mathbf{d_i} \times \mathbf{n}|, \tag{4.6}$$

and which are illustrated in Figure 4.4 as well, then we define the new distance value d_{VCEDT3}, assigned to each background voxel, as follows:

$$d_{VCEDT3} = \begin{cases} d_a + d_v^{grad}, & \text{for } |d_v^{grad}| \leq 0.5 \wedge d_b \leq 0.5 \\ d_{BEDT} + d_v^{ort}, & \text{otherwise.} \end{cases} \tag{4.7}$$

Finally, it is worth mentioning that proposed EDT algorithms, based on the distances d_{VCEDT2} and d_{VCEDT3}, are of a linear time complexity concerning a number of the image voxels. This follows from the fact that the underlying sequential vector

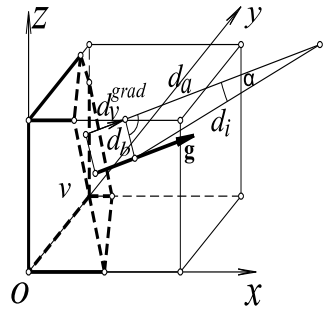


Figure 4.4: Local gradient vector **g**, signed sub-voxel distance d_v^{grad}, distance d_i (4.3) and distances d_a, and d_b (4.6).

propagation-based EDT algorithm [18] is of a linear complexity, i.e., of $\mathcal{O}(N)$ where N is a number of image voxels, while the Algorithm 1 contains no loops and has a constant complexity of $\mathcal{O}(1)$. Besides, it should be emphasized that the estimation of the normal direction requires the distance vector $\mathbf{d}_i$ assigned to each background voxel. In that context, given distance vectors have to be available at each step, we conclude that our framework proposed relates only to vector propagation-based EDT algorithms. In the following we will illustrate the performances of the algorithms utilizing the proposed d_{VCEDT2} and d_{VCEDT3} distances, in comparison both with classical 26SED algorithm [18], using d_{BEDT} (4.3), and the approach (4.4), presented in [91], where computation of d_{VCEDT1} relies only on the voxel coverage information, but not on the normal direction of the object boundary.

The experimental evaluation will relate to the achieved precision and accuracy of the proposed estimation algorithms, as well as their invariance concerning the rotation and translation. Herein under the term precision, we will mean how close the differences between the correct and estimated distances (i.e., errors of estimation) are to each other, while the accuracy will refer to how close the estimation errors are to zero. In other words, the precision will express the degree of repeatability (i.e., reproducibility) of the estimated results when the estimations are repeated under unchanged conditions (less variance among the estimations repeated), even if they are significantly far from their exact values, while accuracy represents the degree of closeness of the estimated distances to their corresponding exact values (less bias among the estimates obtained). In this context, if the differences of the estimated distance values from their corresponding exact values are closer to each other, then we say that the estimate itself is more precise, whereas if the estimated distances are closer to their exact values, then we say that the estimate is more accurate.

4.3 Performance of proposed 3D EDTs

To illustrate the performance of the proposed sub-voxel EDT estimations, we observe 30 different spheres with real-valued diameters in the range from 2 to 62 voxels, and 30 cubes having real-valued edge lengths from 2 to 42 voxels. For each diameter, and edge length, we generate 50 spheres and 50 cubes, respectively, with centers randomly positioned within the voxel, where cubes are additionally rotated randomly (using composition of 3 successive random rotations about the coordinate axes). The objects we observe are digitized in a rectangular grid of the size of $80 \times 80 \times 80$ voxels using Gauss and voxel coverage digitization (following Definition 2.16 and 2.5), respectively. Volume coverage of a voxel is estimated by super-sampling of the boundary voxels by a factor 16, and counting the number of sub-voxel centers covered by the object (as already explained in Definition 2.22).

For computing DTs of the observed test objects, we utilize four above mentioned estimation methods, denoted as BEDT, VCEDT1, VCEDT2, and VCEDT3. The same propagation code is used for BEDT and voxel coverage based estimation methods, where the implementation of the latter ones requires additional consideration of the sub-voxel signed distance, as already described above. For evaluating the performance achieved for each of the EDT algorithms, three quantitative measures are being considered:

1) *root mean square error* (RMSE):

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{x}_i - x_i)^2}\,, \tag{4.8}$$

2) *mean absolute error* (MAE):

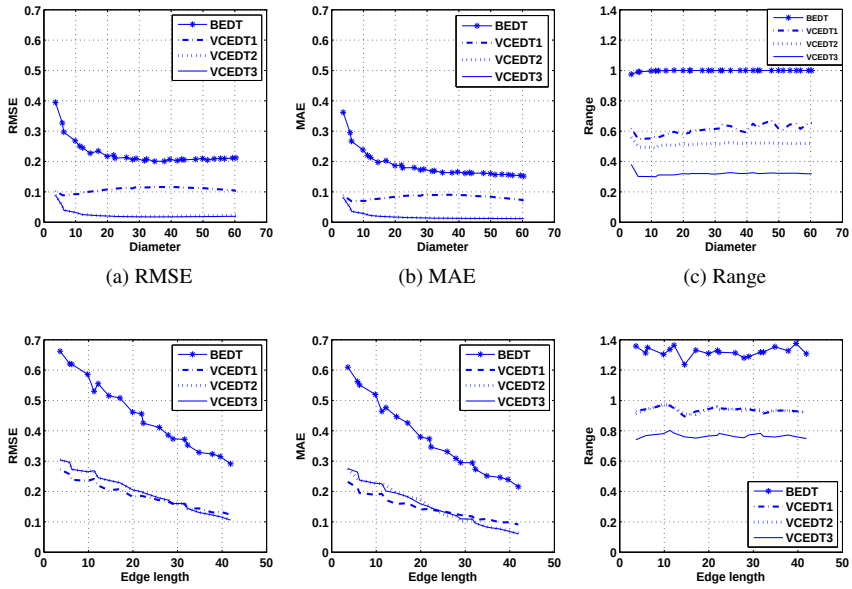
$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{x}_i - x_i|\,, \tag{4.9}$$

3) *empirical range of errors* (Range):

$$\mathit{Range} = \max_{i=1\ldots n}(\hat{x}_i - x_i) - \min_{i=1\ldots n}(\hat{x}_i - x_i)\,, \tag{4.10}$$

where n is a number of image voxels, x_i are the true values of EDT and $\hat{x}_i$ are the estimated EDT values.

For each of the observed size (diameter, or edge length), 50 digitized spheres and 50 digitized cubes, randomly orientated and translated within the digital grid, are generated and above-mentioned measures are computed. The errors computed, measured in voxels, are then averaged over these 50 instances and plotted in Fig. 4.5.



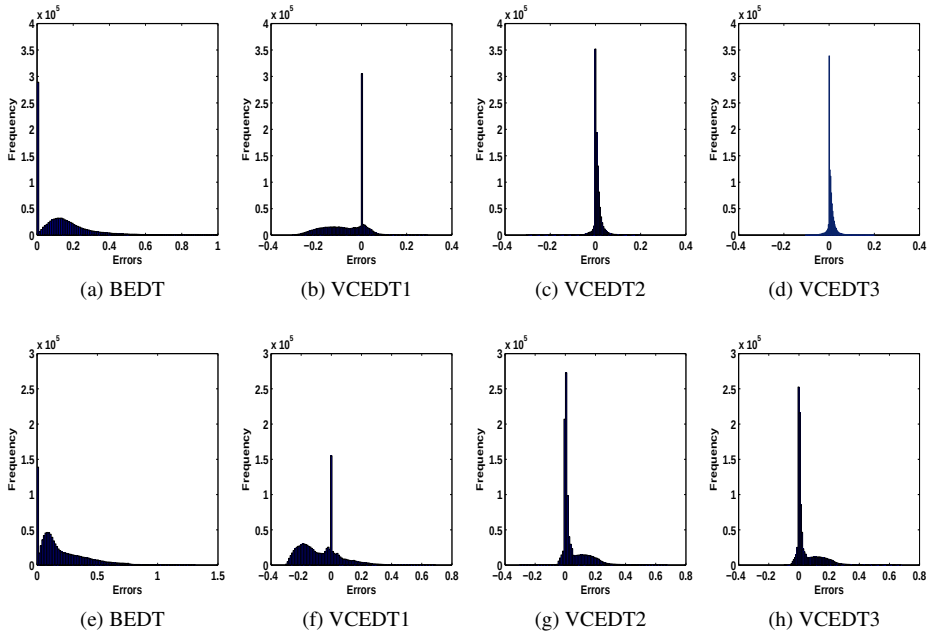
(a) RMSE (b) MAE (c) Range

(d) RMSE (e) MAE (f) Range

Figure 4.5: RMSE, MAE and Range, measured in voxels, of BEDT, VCEDT1 and the proposed VCEDT2, VCEDT3 applied to spheres (first row) and cubes (second row) of increasing sizes.

It can be noticed that the use of coverage values contributes to improvement in accuracy over BEDT. Besides, proposed VCEDT2 and VCEDT3 outperform VCEDT1 method in general, especially in the case of the spheres considered. The improvement achieved for larger cubes is also obvious, while the methods proposed do not provide the performance improvement for cubes with an edge length of less than 20 voxels. Regarding the range values obtained, the reduction achieved by VCDTE3 is visible for all the objects tested.

To provide further experimental evaluation, we consider deviations of the computed EDTs of a randomly positioned sphere with a diameter of 81.8268 voxels, and randomly positioned and additionally rotated cube with an edge length of 51.0938 voxels. To ensure that the observed objects are entirely contained within the rectangular sampling grid, it is necessary to consider the integer grid of a size at least of $\lceil (a * \sqrt{3}) \rceil^3$ voxels where a is edge length of the observed cube. As a result, the underlying rectangular digital grid is now of size of $100 \times 100 \times 100$ voxels. Histograms of errors for these objects are shown in Fig. 4.6, while detailed results are in Table 4.1



(a) BEDT (b) VCEDT1 (c) VCEDT2 (d) VCEDT3

(e) BEDT (f) VCEDT1 (g) VCEDT2 (h) VCEDT3

Figure 4.6: Histograms of errors of BEDT, VCEDT1 and proposed VCEDT2, VCEDT3, when applied to a sphere of diameter 81.8268 (first row) and a cube with edge length 51.0938 (second row).

Table 4.1: RMSE, MAE and Range, measured in voxels, and Relative improvement (regarding MAE) of BEDT, VCEDT1 and proposed VCEDT2, VCEDT3, applied to a sphere with a diameter of 81.8268 voxels.

Methods	BEDT	VCEDT1	VCEDT2	VCEDT3
RMSE (voxels)	0.2016	0.1054	0.0216	0.0181
MAE (voxels)	0.1371	0.0718	0.0115	0.0101
Range (voxels)	0.9999	0.5994	0.5063	0.3042
Rel. improvement	1	1.91	11.92	13.57
Rel. improvement	⁄	1	6.24	7.11

and 4.2, for the sphere and cube, respectively. It can be noticed that noticeable improvement of both precision and accuracy has been achieved in the case of proposed VCEDT2 and VCEDT3 methods, in comparison with BEDT and VCEDT1. Also, it

Table 4.2: RMSE, MAE and Range, measured in voxels, and Relative improvement (regarding MAE) of BEDT, VCEDT1 and proposed VCEDT2, VCEDT3, applied to a cube with edge length of 51.0938 voxels.

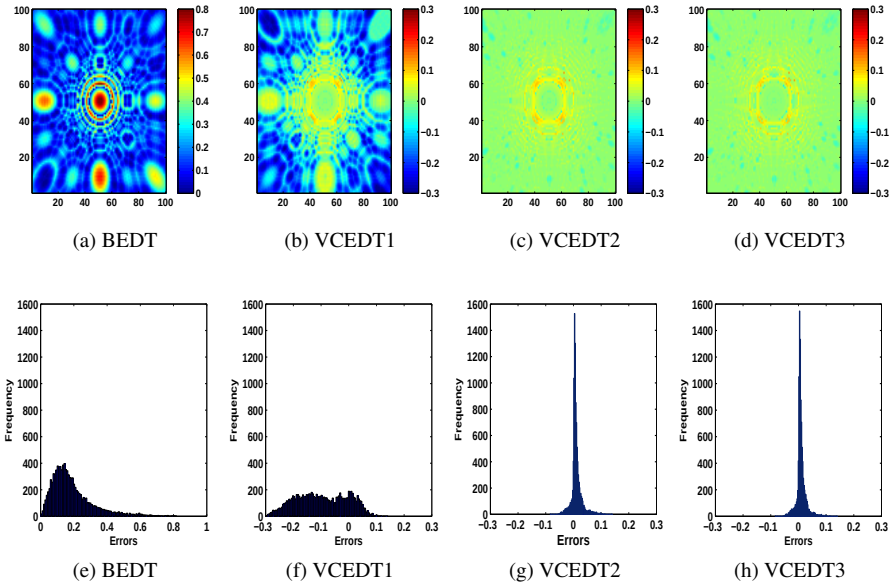
Methods	BEDT	VCEDT1	VCEDT2	VCEDT3
RMSE (voxels)	0.2851	0.1500	0.0954	0.0946
MAE (voxels)	0.2048	0.1173	0.0515	0.0503
Range (voxels)	1.3140	0.9958	0.9813	0.7774
Rel. improvement	1	1.75	3.98	4.07
Rel. improvement	⁄	1	2.28	2.33

Table 4.3: RMSE, Range, MAE, measured in voxels, of the 2D cross-section of error-image with the largest sum of absolute errors, for BEDT, VCEDT1 and proposed VCEDT2, VCEDT3, applied to a sphere.

Methods	BEDT	VCEDT1	VCEDT2	VCEDT3
RMSE	0.2282	0.1335	0.0210	0.0194
MAE	0.1890	0.1089	0.0139	0.0131
Range	0.8180	0.4482	0.2254	0.2254

notices that the results obtained indicate increased accuracy (i.e., reduction in MAE) of approximately 4 to 14 times of the proposed VCEDT2 and VCEDT3 methods, compared to BEDT, as well as of 2 to 7 times concerning VCEDT1. To additionally illustrate the performance of proposed VCEDT2 and VCEDT3 relative to both BEDT and VCEDT1, we have observed how many voxels, in the corresponding distance maps, are assigned the distance values as close as possible to the exact Euclidean distances. In the case of the sphere considered, we have noticed that 46.72%, 66.94%, 99.28% and 99.62% of the voxels are assigned the distance values with accuracy up to ± 0.1 voxels in the BEDT, VCEDT1, VCEDT2 and VCEDT3 distance maps, respectively. Regarding the cube, we have noticed as well that 61.98%, 78.05%, 93.11% and 93.25% of the voxels are assigned the distance values with accuracy up to ± 0.2 voxels, in the corresponding distance maps generated.

For further evaluation, we have observed the errors of the 2D slice of the computed 3D distance maps in the case of the sphere already considered. In the first row of Fig. 4.7, for each of the algorithms considered, are presented the 2D sections with the largest sum of the absolute errors of the distance values assigned to image voxels. For all distances, such presented 2D sections correspond to a plane close to the object boundary. This follows from the fact that the integer vectors, assigned to the voxels near the object boundary, do not align well with the exact normal direction



(a) BEDT (b) VCEDT1 (c) VCEDT2 (d) VCEDT3

(e) BEDT (f) VCEDT1 (g) VCEDT2 (h) VCEDT3

Figure 4.7: First row presents 2D cross-sections of 3D error-image, with the largest sum of absolute errors, for (a) BEDT, (b) VCEDT1, (c) VCEDT2, (d) VCEDT3, applied to a sphere (of diameter of 81.8268). In the second row are present histograms for respective 2D cross-sections shown in the first row.

of the object boundary, as it has been theoretically discussed in the previous section. Histograms of the observed errors for such generated 2D sections are present in the second row of Fig. 4.7, whereas the values of the observed errors for each of the 2D sections are contained in Table 4.3. It can be noticed that the reduction of errors in the case of the proposed VCEDT2 and VCEDT3 estimates are visible, given that the brighter pixels point to higher errors assigned, and, consequently, less accurate algorithms, which is particularly visible in the case of BEDT and VCEDT1 algorithms.

In order to additionally emphasize the improvement of the proposed estimations in terms of rotational and translational invariance achieved, we observe MAE of BEDT, VCEDT1, and proposed VCEDT2, VCEDT3, when they are applied to a randomly positioned and rotated cube with edge length of 50.6802 voxels in a sampling grid of a size of $100 \times 100 \times 100$ voxels. The cube considered is further rotated around the z-axis for angles from 0 to 180 degrees with a step of 5 degrees. For such generated cubes, the MAE as a function of the rotation angle is provided in Fig. 4.8. Absolute reduction in errors, as well as the reduced angular variation of the proposed

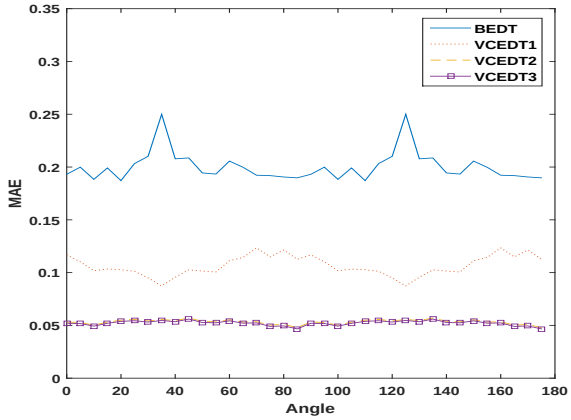


Figure 4.8: MAE of BEDT, VCEDT1 and proposed VCEDT2, VCEDT3, when applied to a randomly positioned cube with edge length of 50.6802 voxels which is rotated around the z-axis from 0 to 180 degrees with a step of 5 degrees.

Table 4.4: MAE, measured in voxels, of BEDT, VCEDT1, VCEDT2, and VCEDT3, when applied to a randomly positioned cube with edge length of 50.6802 voxels which is rotated (top rows) around the z-axis from 0 to 180 degrees with a step of 5 degrees, or translated (bottom rows) to 20 random positions.

Methods	BEDT	VCEDT1	VCEDT2	VCEDT3
Rot. MAE (average)	0.2003	0.1065	0.0533	0.0522
Rot. MAE (stddev)	0.0142	0.0095	0.0022	0.0023
Transl. MAE (average)	0.1955	0.0941	0.0574	0.0563
Transl. MAE (stddev)	0.0066	0.0014	0.0015	0.0015

VCEDT2 and VCEDT3, compared to both BEDT and VCEDT1. Further evaluation is performed for the cubes randomly translated within one voxel can be noticed. The results obtained, contained in Table 4.4, are similar to those already presented, except in the case of translational variance of VCEDT1 which is equal to ones achieved for VCEDT2 and VCEDT3. In order to evaluate numerically how the generated distances vary through rotation and translation, in Table 4.4 we show the average MAE and standard deviation of MAE over all the observed rotations and translations. It can also be noticed that a considerable decrease of both rotational and translational variability, in the case of the proposed VCEDT2 and VCEDT3 estimates, is achieved.

To better understand the behaviour of the proposed sub-voxel EDT estimates,

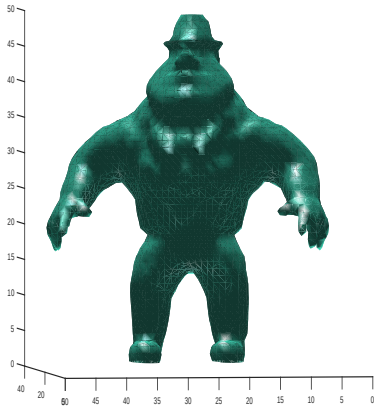
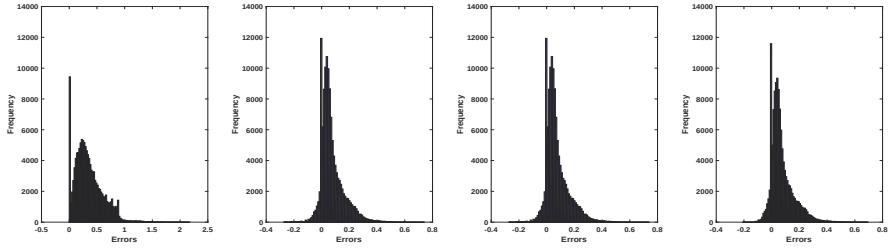


Figure 4.9: Al Capone object.

we have illustrated their performances on a more complex shape. As a reference shape for this experiment, we utilize the original Al Capone object[1] visualized in Fig. 4.9, while for computing the ground truth distance values we utilize the method proposed in [99]. Histograms of errors for the observed BEDT, VCEDT1, VCEDT2, and VCEDT3 algorithms are shown in Fig. 4.10, while the corresponding performance measures are given in Table 4.5. It is easy to notice that the accuracy improvement resulting from the use of voxel coverage information is visible, whereas the improvement relative to VCEDT1 is not as pronounced as for the simpler objects, as shown in the previous experiments.

At the end of the experimental section, it should be said something about the performance of the EDT estimation algorithms in terms of their execution time. Herein we will observe the computational complexity of the algorithms discussed to the number of image voxels. The plots presenting the execution time, measured in seconds, for each algorithm considered to the image size are shown in Fig. 4.11. It can be noticed that the presented plots clearly illustrate that the algorithms discussed are of linear time complexity. This is also consistent with the theoretical consideration, discussed in the previous section, that the algorithms presented are of linear complexity. Also, using the look-up table of pre-computed distance values provided by Algorithm 1, the algorithm execution can be additionally accelerated if it is necessary.

[1]The original $450 \times 450 \times 450$ voxels Al Capone, submitted by Eric Remy, is available on the IAPR-TC18 webpage (http://www.tc18.org).



(a) BEDT (b) VCEDT1 (c) VCEDT2 (d) VCEDT3

Figure 4.10: Histograms of errors for BEDT, VCEDT1, VCEDT2, and VCEDT3, computed for $50 \times 50 \times 50$ Al Capone object.

Table 4.5: RMSE, MAE and Range, measured in voxels, of BEDT, VCEDT1, VCEDT2, and VCEDT3, computed for $50 \times 50 \times 50$ Al Capone object.

Methods	BEDT	VCEDT1	VCEDT2	VCEDT3
RMSE	0.1858	0.0124	0.0119	0.0118
MAE	0.3510	0.0819	0.0771	0.0760
Range	2.1773	1.0068	1.0128	0.9037

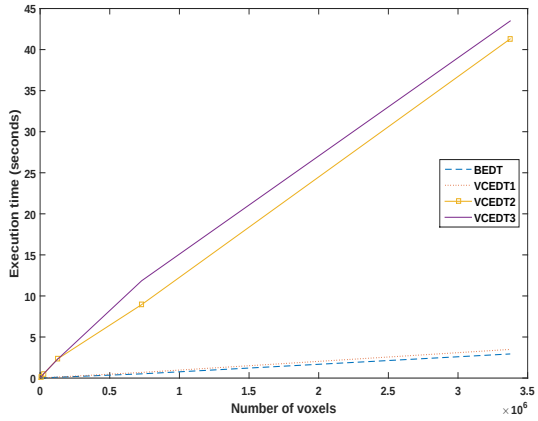


Figure 4.11: Measured execution time, in seconds, of BEDT, VCEDT1, VCDET2 and VCEDT3, applied to Al Capone object of size $18^3, 30^3, 50^3, 90^3$, and 150^3 voxels, as a function of the number of image voxels.

Chapter 5

Hexagonality as a new shape descriptor of the object

In this chapter, we consider a hexagonality as a new shape-based descriptor of the object and also present a new measure which evaluates how much a given shape is hexagonal. As one of the interesting, and also a specific aspect of the shape, the hexagonality has a clear geometric meaning, and, as such, it can be defined by adopting some usual schemes for defining other shape descriptors. Probably the most obvious approach for defining a general class of shape measures is described in cite Rosin2008 and consists of fitting the model to the considered shape and then expressing the shape measure by the level of their match. Consequently, for example, one of the possibilities for defining the shape hexagonality can be described as follows: let fit in some way appropriately chosen hexagon $\mathcal{F}H(S)$ to a shape S we are measuring, and evaluate hexagonality of S relative to the fitted hexagon $\mathcal{F}H(S)$. One the natural choice for a fitted hexagon can be the hexagon whose centroid coincides the shape centroid, and having the area equals the area of shape. Taking this into account, a shape hexagonality measure can be defined as

$$\mathcal{H}_{fit}(S) = \frac{Area\,(S \cap \mathcal{F}H(S))}{Area\,(S \cup \mathcal{F}H(S))}, \tag{5.1}$$

where $\mathcal{F}H(S)$ denotes a hexagon rotated around its centroid to provide that the area of $S \cap \mathcal{F}H(S)$ reaches its maximal possible value. Defined that way, hexagonality measure $\mathcal{H}_{fit}(S)$ satisfies the following important and, in many applications, desirable properties:

- The measured hexagonality $\mathcal{H}_{fit}(S)$ is a number from $[0,1]$;
- The measured hexagonality $\mathcal{H}_{fit}(S)$ is equal to 1 if and only if S is a hexagon;

- There exist shapes with measured hexagonality $\mathcal{H}_{fit}(S)$ equals 0;
- The measured hexagonality is invariant under similarity transformations.

It can be said that most of the properties represent natural requirements that any shape descriptor should satisfy. However, among the properties listed above, the first and third ones deserve special attention and additional consideration. To be more precise, according to the properties stated in the first and third items, we have that there exist shapes, with the non-zero area, whose measured hexagonality amounts 0. Examples of such shapes are the shapes having no intersection with the corresponding fitted hexagon (for example, see the last shape presented in Fig. 5.6). This further implies that 0 is the lowest possible value for $\mathcal{H}_{fit}(S)$, implying also that the interval $[0,1]$ cannot be made as narrow as possible. As such, this property is not so desirable, given that our initial intention was to design a new shape based measure which should answer how much a given shape differs from a hexagon. This further motivates us to consider another measure of hexagonality that has all the above-mentioned properties, with the exception that for each non-zero area shape the assigned hexagonality should be greater than 0. In this chapter, we introduce a new hexagonality measure which ranges through the interval $(0,1]$ and assigns the largest value amounts 1 if and only if the shape we are measuring is a hexagon. Also, it is invariant concerning rotation, translation and scaling transformations.

Additional motivation for our research stems from the tasks of computer vision and image analysis, given the new hexagonality measure has several useful consequences which are also discussed and experimentally evaluated in this chapter. As a first consequence, we present a new method for calculating shape orientation, where the shape orientation is determined by an angle (i.e., direction), minimizing the integral in (5.21). Effectiveness of the new approach for determining the shape orientation has been also demonstrated and compared with the approach for computing the shape orientation based on the axis of the least second moment of inertia [152]. Such an introduced axis represents the line minimizing the integral of the squared distances of all shape points to the line, and orientation of a shape is then defined as a slope of such a line. This approach is also referred to as *moment-based*, given that all the second-order shape's moments, as given in (5.2), are used for its computing. Besides, the new method for computing the shape hexagonality provides as well a new shape elongation measure. Indeed, taking into account that a new hexagonality measure evaluates how much the shape considered differs from a hexagon, it is natural to define shape elongation as a ratio of the lengths of the longer and shorter semi-axis of the appropriately associated hexagon. Such a defined elongation measure has several desirable properties that are verified both from a theoretical point of view and experimentally using several illustrative examples. To additionally demonstrate the behaviour of the new elongation measure, we also compared it with the standard elongation measure based on the shape's moments, which has been derived from the

already mentioned standard method for computing shape orientation. All these desirable properties are also theoretically proven. This can be particularly beneficial in certain applications, given that the behaviour of such a measure can be predicted in advance. Several experiments, performed on both synthetic and real image data, are shown to confirm the theoretical observations and illustrate the behaviour of the new shape-based measures. Given that new measures are not designed for a particular application, they can be applied in different object analysis tasks. Several experiments relating to three well-known image datasets such as MPEG-7 CE-1 [73], Swedish Leaf [151], and Galaxy Zoo [92] datasets, are also provided to illustrate effectiveness and benefits of the new shape measures in a variety of object classification tasks.

The chapter is organized as follows. The basic terms necessary for deriving the main result of the chapter are in the next section. Section 5.2 introduces a new hexagonality measure and proves several desirable properties of it. The experiments, performed on synthetic and real image data, that illustrate the behaviour of the new hexagonality measure and also provide its comparison with $\mathcal{H}_{fit}(S)$, are in Section 5.3. Several consequences of the new hexagonality measure are in Section 5.4. A few illustrative examples and theoretical considerations related to these consequences are also included in Section 5.4. Section 5.5 provides several experiments illustrating applicability and usefulness of the new hexagonality measure in different shape-based object analysis tasks.

5.1 Definitions and assumptions

Herein we recall some elementary definitions necessary to derive the main result of the chapter. We also introduce several assumptions that do not restrict research presented in this chapter, but which are necessary to provide that research is theoretically well-founded. As it has already been mentioned, the shape is one of the basic properties of the object along with texture and color. In that context, the shape is represented by bounded (not necessarily connected) planar region, or as a set of the black pixels on the black-white digital images.

In our derivation of the main result, we rely on moments assigned to the shape considered. The (p,q)-*moment* $m_{p,q}(S)$ of a planar shape S is defined as follows

$$m_{p,q}(S) = \iint_S x^p y^q dxdy \tag{5.2}$$

and has the order $p+q$. The basic features of a shape such as the size (i.e., area) and position (i.e., centroid) can be computed using the moments of order not greater than one. Precisely, the zeroth-order moment $m_{0,0}(S)$ equals the area of S, while the centroid of S is defined as $\left(\frac{m_{1,0}(S)}{m_{0,0}(S)}, \frac{m_{0,1}(S)}{m_{0,0}(S)}\right)$. In the following, unless otherwise stated, we will assume that all the shapes considered have the centroid coincident with the

origin, i.e., $m_{1,0}(S) = 0$ and $m_{0,1}(S) = 0$. Given that a shape of the object does not change under the translation transformation, this assumption shall not be considered a restriction in the shape-based object analysis tasks. We also assume that two shapes are equal if and only if the area of their set difference is equal to zero. For instance, an open d_1-distance disc (i.e., an open square) $S_1 = \{(x,y)\big||x|+|y|<1\}$ and the closed d_1-distance disc (i.e., the closed square) $S_2 = \{(x,y)\big||x|+|y|\leq 1\}$ are of the same shape.

Further, for our research, the first two Hu moment invariants derived from the second-order moments are of particular significance. These quantities represent two best-known and also most widely used Hu moment invariants among all the introduced in [66], primarily because of their computational complexity, given that only the second-order moments are needed for their computation. For a given shape S, whose centroid coincides with the origin, the first two Hu moment invariants are defined as follows

$$\mathcal{H}_1(S) = \frac{1}{m_{0,0}(S)^2} \cdot \left(m_{2,0}(S) + m_{0,2}(S)\right), \tag{5.3}$$

$$\mathcal{H}_2(S) = \frac{1}{m_{0,0}(S)^4} \cdot \left(\left(m_{2,0}(S) - m_{0,2}(S)\right)^2 + 4m_{1,1}(S)^2\right). \tag{5.4}$$

It is worth noting that all Hu moment invariants are the geometric moment invariants as well, given that they can be generated using the corresponding geometric primitives (i.e., invariants) [172]. Because of this, some of the Hu moment invariants can also be used to measure certain shape descriptors, which can be understood as their advantage in some applications.

Finally, a new shape hexagonality measure will be derived by using d_1-distance, which is, for any two given planar points $A(x_1,y_1)$ and $B(x_2,y_2)$, defined in (3.3) as follows:

$$d_1(A,B) = d_1((x_1,y_1),(x_2,y_2)) = |x_1 - x_2| + |y_1 - y_2|. \tag{5.5}$$

It is easy to verify that the set of all the points whose sum of their d_1-distances from two fixed points (also called *foci*) is smaller than a given constant a is a *hexagon*. Without loss of generality, we will herein observe a hexagon having the centroid coincident with the origin, and whose foci have the coordinates $(c,0)$ and $(-c,0)$. Such a defined hexagon has semi-axes parallel to the coordinate axes, and as such has a form

$$Hex(a,c) = \{(x,y) \;\big|\; |x-c| + |y| + |x+c| + |y| \leq 2 \cdot a\} \tag{5.6}$$

where a is a given constant. Moreover, such a hexagon is also said to be an *isothetic hexagon*. In a special case, when foci coincide with the origin, or equivalently for $c = 0$, the formula in (5.6) defines a square (i.e., disc in terms of d_1-distance), herein denoted with $S(a)$, as follows

$$S(a) = Hex(a,0) = \{(x,y) \;\big|\; |x| + |y| \leq a\}. \tag{5.7}$$

Thanks to this fact, a square can be considered as a degenerative hexagon, which will enable that a new shape hexagonality measure is well-defined, and its derivation mathematically correct as well.

5.2 New hexagonality measure

In this section, we provide the theoretical framework that enables us to derive the main result of the chapter. Considering these theoretical observations, we will define a new shape hexagonality measure and describe several desirable properties of it. In that context, of particular interest for our derivation will be the quantity defined as

$$\min_{\theta\in[0,2\pi)} \iint_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \tag{5.8}$$

where S_θ denotes rotation of a shape S for an angle θ around its centroid. We will show that such a considered quantity achieves the lowest possible value if and only if S is a hexagon. This result will further lead us to a new shape hexagonality measure. Before we start with our derivation, we first define an auxiliary hexagon $Hex(S)$ for a given shape S as follows:

$$Hex(S) = \{(x,y) \mid |x-c(S)|+|y|+|x+c(S)|+|y| \leq 2\cdot a(S)\}, \tag{5.9}$$

where the associated parameters $a(S)$ and $c(S)$ are defined as follows:

$$a(S) = \left(Area(S)\cdot\left(3\cdot\mathcal{H}_1(S)+3\cdot\sqrt{\mathcal{H}_2(S)}+1/2\right)/2\right)^{1/2}, \tag{5.10}$$

$$c(S) = \left(Area(S)\cdot\left(3\cdot\mathcal{H}_1(S)+3\cdot\sqrt{\mathcal{H}_2(S)}-1/2\right)/2\right)^{1/2}. \tag{5.11}$$

For such a defined hexagon $Hex(S)$, it holds the following:

- $Hex(S)$ depends only on a given shape S, and it does not change if S is rotated for an arbitrary angle. This comes from the fact that both $a(S)$ and $c(S)$ are computed from the first two Hu moment invariants and the shape area, which are rotational invariants as well.

- The associated parameters $a(S)$ and $c(S)$ of the auxiliary hexagon $Hex(S)$, as given in (5.10) and (5.11), are set in the way that the area of $Hex(S)$ is equal to area of S, i.e., $Area(Hex(S)) = Area(S) = 2\cdot\left(a(S)^2 - c(S)^2\right)$, and also that the first two Hu moment invariants of $Hex(S)$ and S are equal, i.e., $\mathcal{H}_1(S) = \mathcal{H}_1(Hex(S))$ and $\mathcal{H}_2(S) = \mathcal{H}_2(Hex(S))$.

Note 5.1 *It is easy to notice that the formulas in* (5.10) *and* (5.11) *make sense if and only if* $\mathcal{H}_1(S)+\sqrt{\mathcal{H}_2(S)}\geq 1/6$. *However, given that there exist shapes for which this relation breaks down (e.g., for a disc it equals* $1/(2\pi)$*), it makes sense, in such cases, to set* $c(S)=0$, *while the other associated parameter* $a(S)$ *is computed in the way that the areas of* $Hex(S)$ *and* S *are equal, i.e.,* $a(S)=\sqrt{Area(S)/2}$. *It should be noticed that this is not a restriction for defining a new hexagonality measure since, in such cases, the corresponding auxiliary hexagon is a square* $S(a)$, *i.e., a degenerative hexagon defined in* (5.7).

Now, we start with the derivation of the main result, where, in some statements, we follow the approach used in [122, 161]. First, the following theorem applies.

Theorem 5.1 *Let a shape* S, *whose centroid coincides with the origin, be given, and also let* S_θ *be a shape* S *rotated by an angle* θ *around the origin. Then the following statements are true:*

$$\frac{\iint\limits_S (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy}{a(S)^3-c(S)^3} \geq \frac{8}{3}; \tag{5.12}$$

$$\frac{\iint\limits_S (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy}{a(S)^3-c(S)^3} = \frac{8}{3} \Leftrightarrow S=Hex(S), \tag{5.13}$$

$$\frac{\min\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy}{a(S)^3-c(S)^3} = \frac{8}{3} \Leftrightarrow S \text{ is a hexagon.} \tag{5.14}$$

Proof 5.1 *Let* S *be a shape satisfying the conditions of the theorem. Also, let denote with* $Hex(S)$ *the auxiliary isothetic hexagon, centered at the origin, with the vertices* $(-a(S),0)$, $(-c(S),-a(S)+c(S))$, $(c(S),-a(S)+c(S))$, $(a(S),0)$, $(c(S),a(S)-c(S))$ *and* $(-c(S),a(S)-c(S))$. *It is easy to check that the areas of* S *and* $Hex(S)$, *as well as the areas of* $S\setminus Hex(S)$ *and* $Hex(S)\setminus S$.

From the definition of $Hex(S)$, *it holds that the sum of* d_1*-distances of the points from* $Hex(S)\setminus S$ *to foci* $(c(S),0)$ *and* $(-c(S),0)$ *is smaller than the sum of* d_1*-distances of the points from* $S\setminus Hex(S)$ *to* $(c(S),0)$ *and* $(-c(S),0)$, *i.e., if* $(u,v)\in S\setminus Hex(S)$

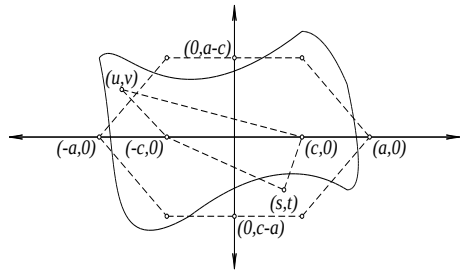


Figure 5.1: Sum of d_1-distances of each point (s,t) from $Hex(a,c) \setminus S$ to foci $(c,0)$ and $(-c,0)$ is smaller than the sum of d_1-distances of any point (u,v) from $S \setminus Hex(a,c)$ to $(c,0)$ and $(-c,0)$.

and $(s,t) \in Hex(S) \setminus S$ *then the following applies:*

$$|u-c(S)|+|v|+|u+c(S)|+|v| \\ > |s-c(S)|+|t|+|s+c(S)|+|t|. \tag{5.15}$$

This further implies that the following equality

$$\iint\limits_{S \setminus Hex(S)} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\ - \iint\limits_{Hex(S) \setminus S} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\ = Area(S \setminus Hex(S)) \cdot (|u-c(S)|+|v|+|u+c(S)|+|v|) \\ - Area(Hex(S) \setminus S) \cdot (|s-c(S)|+|t|+|s+c(S)|+|t|), \tag{5.16}$$

holds true for some $(u,v) \in S \setminus Hex(S)$ *and some* $(s,t) \in Hex(S) \setminus S$. *Given* $Area(Hex(S) \setminus S) = Area(S \setminus Hex(S)) \geq 0$, *the last equality along with* (5.15) *gives the following:*

$$\iint\limits_{S \setminus Hex(S)} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \geq \\ \iint\limits_{Hex(S) \setminus S} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy. \tag{5.17}$$

Now, we establish the proof of (5.12)*:*

$$\begin{aligned}
&\iint_S (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \;= \\
&\quad= \iint_{S\setminus Hex(S)} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\
&\qquad+ \iint_{S\cap Hex(S)} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\
&\quad\geq \iint_{Hex(S)\setminus S} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\
&\qquad+ \iint_{S\cap Hex(S)} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\
&\quad= \iint_{Hex(S)} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\
&\qquad= \frac{8}{3}\cdot\left(a(S)^3-c(S)^3\right).
\end{aligned}$$

Further, notice that inequality in (5.17) *is strict if and only if* $Area(S\setminus Hex(S)) = Area(Hex(S)\setminus S) > 0$*, i.e., if and only if the shape S is different from a hexagon. This completes the proof of* (5.13)*, since the equality in* (5.17) *holds true if and only if* $Area(S\setminus Hex(S)) = Area(Hex(S)\setminus S) = 0$*, i.e., if and only if the shapes S and* $Hex(S)$ *are the same.*

To prove (5.14)*, let first notice that* $\iint_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy$ *depends on the orientation of the shape S, i.e., it changes over all rotations of S around the origin for an angle varying through the interval* $[0,2\pi)$*. Instead, we consider the minimum of this quantity for all rotations* S_θ *of the shape S around the origin. Therefore, let denote with* θ_0 *the angle minimizing such a quantity, i.e.,*

$$\begin{aligned}
&\iint_{S_{\theta_0}} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \\
&\qquad= \min_{\theta\in[0,2\pi)} \iint_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy. \qquad (5.18)
\end{aligned}$$

Now, if we suppose that the left side of (5.14) *is true, i.e.,*

$$\frac{\min\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy}{a(S)^3-c(S)^3} = \frac{8}{3},$$

then we obtain

$$\frac{\iint\limits_{S_{\theta_0}} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy}{a(S)^3-c(S)^3} = \frac{8}{3}. \tag{5.19}$$

This completes the proof, since the equality in (5.19)*, according to* (5.13)*, implies that shapes $S(\theta_0)$ and $Hex(S)$ coincide, i.e., S must be a hexagon.* □

Now, exploiting the observations and results of Theorem 5.1, we have that the quantity

$$\frac{\min\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy}{a(S)^3-c(S)^3}$$

does not depend on the orientation of S (i.e., it is invariant to rotation), and reaches the minimum $8/3$ if and only if S is a hexagon. Based on this, we can conclude that the quantity

$$\frac{8}{3}\cdot\frac{a(S)^3-c(S)^3}{\min\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy} \tag{5.20}$$

can be used as a new shape measure. Herein such a defined shape measure is named a *shape hexagonality measure*, and will be denoted as $\mathcal{H}(S)$. Now, we give a formal definition for $\mathcal{H}(S)$.

Definition 5.1 *Let a shape S, having the centroid coincident with the origin, be given, and let $a(S)$ and $c(S)$ be the associated parameters, defined as in* (5.10) *and* (5.11)*, or, if necessary, as in Note 5.1. The shape hexagonality measure of a given shape S, denoted as $\mathcal{H}(S)$, is defined as*

$$\mathcal{H}(S) = \frac{8}{3}\cdot\frac{a(S)^3-c(S)^3}{\min\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy} \tag{5.21}$$

with S_θ denoting shape S rotated by θ around its centroid.

Such a defined measure satisfies the basic properties of the hexagonality measure in (5.1), with small modification regarding the first, and consequently the third property, which has been already indicated to represent weakness for each shape measure. To formalize this, the following theorem, based on the results and arguments of Theorem 5.1, summarizes several properties of the new $\mathcal{H}(S)$ measure.

Theorem 5.2 *Let a shape S, whose centroid is coincident with the origin, be given. The hexagonality measure $\mathcal{H}(S)$ satisfies the following properties:*

(a) $\mathcal{H}(S) \in (0,1]$, for all shapes S;

(b) $\mathcal{H}(S) = 1 \Leftrightarrow$ S is a hexagon;

(c) $\mathcal{H}(S)$ is invariant with respect to translation, rotation and scaling transformations.

Proof 5.2 *The first two statements follow directly from* (5.12) *and* (5.14) *provided by Theorem 5.1. The third statement comes from the fact that the quantities $a(S)$ and $c(S)$, needed for the computation of $\mathcal{H}(S)$, and the quantity in the denominator in* (5.21) *are rotational invariants by their construction, implying that $\mathcal{H}(S)$ is rotational invariant as well. Invariance to translation follows from the fact that the centroid of S is always coincident with the origin. To prove the scaling invariance of $\mathcal{H}(S)$, let suppose that S is scaled by some factor λ to shape $\lambda \cdot S = \{(\lambda \cdot x, \lambda \cdot y) \mid (x,y) \in S\}$. Then, it holds:*

$$\begin{aligned} Area(\lambda \cdot S) &= \iint_{\lambda \cdot S} dxdy = \iint_{S} \lambda^2 dxdy \\ &= \lambda^2 \cdot Area(S), \end{aligned} \tag{5.22}$$

$$\begin{aligned} a(\lambda \cdot S) &= \left(Area(\lambda \cdot S) \cdot \left(3\mathcal{H}_1(\lambda \cdot S) + 3\sqrt{\mathcal{H}_2(\lambda \cdot S)} + 1/2 \right) /2 \right)^{1/2} \\ &= \left(\lambda^2 \cdot Area(\lambda \cdot S) \cdot \left(3\mathcal{H}_1(\lambda \cdot S) + 3\sqrt{\mathcal{H}_2(\lambda \cdot S)} + 1/2 \right) /2 \right)^{1/2} \\ &= \lambda \cdot a(S), \end{aligned} \tag{5.23}$$

$$\begin{aligned} c(\lambda \cdot S) &= \left(Area(\lambda \cdot S) \cdot \left(3\mathcal{H}_1(\lambda \cdot S) + 3\sqrt{\mathcal{H}_2(\lambda \cdot S)} - 1/2 \right) /2 \right)^{1/2} \\ &= \left(\lambda^2 \cdot Area(S) \cdot \left(3\mathcal{H}_1(S) + 3\sqrt{\mathcal{H}_2(S)} - 1/2 \right) /2 \right)^{1/2} \\ &= \lambda \cdot c(S), \end{aligned} \tag{5.24}$$

$$\min_{\theta\in[0,2\pi)} \iint\limits_{\lambda\cdot S_\theta} (|x-c(\lambda\cdot S)|+|y|+|x+c(\lambda\cdot S)|+|y|)\,dxdy$$

$$= \min_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|\lambda\cdot x-\lambda\cdot c(S)|+|\lambda\cdot y|+|\lambda\cdot x+\lambda\cdot c(S)|+|\lambda\cdot y|)\cdot\lambda^2\,dxdy$$

$$= \lambda^3\cdot \min_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)\,dxdy, \tag{5.25}$$

and, therefore,

$$\mathcal{H}(\lambda\cdot S) =$$

$$= \frac{8}{3}\cdot \frac{a(\lambda\cdot S)^3-c(\lambda\cdot S)^3}{\lambda^3\cdot \min\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)\,dxdy}$$

$$= \frac{8}{3}\cdot \frac{a(S)^3-c(S)^3}{\min\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)\,dxdy}$$

$$= \mathcal{H}(S). \tag{5.26}$$

This ends the proof of the theorem, given that (5.26) *proves that* $\mathcal{H}(S)$ *is a scaling invariant as well.* □

Finally, we should say something about the algorithm for numerical computation of $\mathcal{H}(S)$. In that context, Definition 5.1 provides a simple and reasonably efficient algorithm consisting of rotation and considering each shape point. Precisely, we consider all rotations of S around the origin, and compute the integral in the denominator in (5.21) while angle θ varies over $[0,2\pi)$. To accelerate the computation of this integral, the appropriate shapes moments, as given in (5.2), can be computed for a

suitable partition of S_θ specified as follows:

$$\begin{aligned}
S_1(\theta) &= \{(x,y) \in S_\theta \mid x < -c(S),\ y < 0\}, \\
S_2(\theta) &= \{(x,y) \in S_\theta \mid x < -c(S),\ y \geq 0\}, \\
S_3(\theta) &= \{(x,y) \in S_\theta \mid -c(S) \leq x < c(S),\ y < 0\}, \\
S_4(\theta) &= \{(x,y) \in S_\theta \mid -c(S) \leq x < c(S),\ y \geq 0\}, \\
S_5(\theta) &= \{(x,y) \in S_\theta \mid x \geq c(S),\ y < 0\}, \\
S_6(\theta) &= \{(x,y) \in S_\theta \mid x \geq c(S),\ y \geq 0\}.
\end{aligned} \tag{5.27}$$

Exploiting this, the quantities considered can be evaluated as follows:

$$\begin{aligned}
&\iint_{S(\theta)} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy = \\
&= \iint_{S_1(\theta)} (-2\cdot x-2\cdot y)dxdy + \iint_{S_2(\theta)} (-2\cdot x+2\cdot y)dxdy \\
&+ \iint_{S_3(\theta)} (2\cdot c(S)-2\cdot y)dxdy + \iint_{S_4(\theta)} (2\cdot c(S)+2\cdot y)dxdy \\
&+ \iint_{S_5(\theta)} (2\cdot x-2\cdot y)dxdy + \iint_{S_6(\theta)} (2\cdot x+2\cdot y)dxdy \\
&= -2\cdot m_{1,0}(S_1(\theta)) - 2\cdot m_{0,1}(S_1(\theta)) \\
&-2\cdot m_{1,0}(S_2(\theta)) + 2\cdot m_{0,1}(S_2(\theta)) \\
&+2\cdot c(S)\cdot m_{0,0}(S_3(\theta)) - 2\cdot m_{0,1}(S_3(\theta)) \\
&+2\cdot c(S)\cdot m_{0,0}(S_4(\theta)) + 2\cdot m_{0,1}(S_4(\theta)) \\
&+2\cdot m_{1,0}(S_5(\theta)) - 2\cdot m_{0,1}(S_5(\theta)) \\
&+2\cdot m_{1,0}(S_6(\theta)) + 2\cdot m_{0,1}(S_6(\theta))
\end{aligned} \tag{5.28}$$

Following this, it is enough to compute only the zeroth- and first-order moments of the subsets in (5.27), instead of $\iint\limits_{S_\theta} (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy$, making

the computation of $\mathcal{H}(S)$ additionally simpler and faster. Besides, if a continuous shape S is given in a discrete (i.e., digital) space with its Gaussian digitization $\mathcal{D}_{\mathcal{G}}(S)$ (according to Definition 2.16), our approach is based on working as long as possible in the continuous domain, and therefore with the continuous quantities, and delay the effects of discretization (i.e., digitization) as long as possible till the end of the process of computation. Then, the integral $\iint\limits_S (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy$ is approximated as

$$\iint\limits_S (|x-c(S)|+|y|+|x+c(S)|+|y|)dxdy \approx$$

$$\approx \sum_{(i,j)\in\mathcal{D}_{\mathcal{G}}(S)} (|i-c(\mathcal{D}_{\mathcal{G}}(S))|+|j|+|i+c(\mathcal{D}_{\mathcal{G}}(S))|+|j|), \quad (5.29)$$

where the associated parameters $a(S)$ and $c(S)$ are given, respectively, by their approximations $a(\mathcal{D}_{\mathcal{G}}(S))$ and $c(\mathcal{D}_{\mathcal{G}}(S))$ (according to (5.10) and (5.11), or Note 5.1, if it is necessary), while the moments $m_{p,q}(S)$, given in (5.2), are evaluated as follows:

$$m_{p,q}(S) = \iint\limits_S x^p y^q dxdy \approx \sum_{(i,j)\in\mathcal{D}_{\mathcal{G}}(S)} i^p \cdot j^q. \quad (5.30)$$

It can be noticed that the approximation in (5.30) is simple and easy to compute, given that only multiplications and summations are necessary, and also provides a very accurate estimation of the continuous shapes moments [79]. Regarding this, in the case of the digital shapes (i.e., digital images), the latter two approximations are of particular importance to computing a new hexagonality measure $\mathcal{H}(S)$.

5.3 Experiments illustrating $\mathcal{H}(S)$ behaviour

This section presents several experiments aimed to illustrate the behaviour of the new hexagonality measure $\mathcal{H}(S)$. These experiments, performed on both synthetic and real image data, are designed to provide a better understanding of the new hexagonality measure, and to verify the theoretical observations and results already proven in the previous section as well. Also, we provide several experiments to compare the behaviour of $\mathcal{H}(S)$ with initial hexagonality measure $\mathcal{H}_{fit}(S)$, as defined in (5.1). The first subsection concerns the synthetic examples, while the second one contains experimental study on real image data taken from several modern image datasets.

(a) 0.9802 (b) 0.9934 (c) 0.9845 (d) 0.9972 (e) 1.0000

Figure 5.2: Shapes defined by (5.31) for increasing value of ε, and their assigned $\mathcal{H}(S)$ values (given below each shape related).

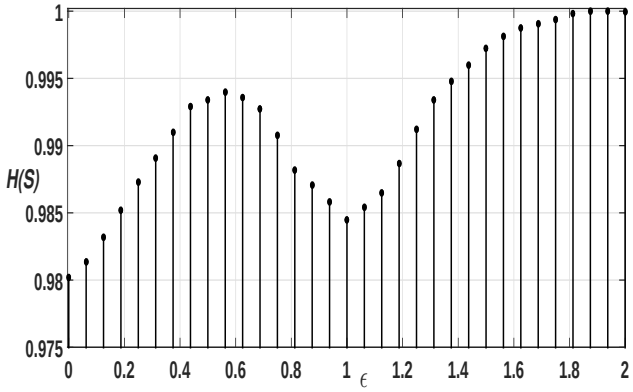


Figure 5.3: Plot of computed $\mathcal{H}(S)$ values of shapes defined in (5.31) for increasing value of parameter ε.

5.3.1 Illustrations of $\mathcal{H}(S)$ behaviour on synthetic examples

First experiment: This experiment illustrates the behaviour of the new hexagonality measure $\mathcal{H}(S)$ as the shape considered is modified through continuous variation from a rectangle to a circle, and then to a hexagon. Several examples of such generated shapes, defined implicitly as

$$S(a,c;\varepsilon) = \left\{ (x,y) \,\middle|\, \left|\frac{x-c}{2a}\right|^{2/\varepsilon} + \left|\frac{x+c}{2a}\right|^{2/\varepsilon} + 2\left|\frac{y}{2a}\right|^{2/\varepsilon} \leq 1 \right\}, \qquad (5.31)$$

are given together with their computed $\mathcal{H}(S)$ values in Fig. 5.2. Such defined shapes change from a rectangle (for $\varepsilon = 0$) through a disc (for $\varepsilon = 1$) to a hexagon (for $\varepsilon = 2$). The results obtained do not contradict our expectations, and they are also consistent with the theoretical observations that have been already discussed in

(a) 1.0000 (b) 0.9860 (c) 0.9723 (d) 0.9589 (e) 0.9459

Figure 5.4: Shapes obtained for increasing amount of added salt and pepper noise to a perfect hexagon given in (a). Computed $\mathcal{H}(S)$ values are given below each shape related. Noise probabilities added to shapes in (b), (c), (d) and (e) are 0.125, 0.25, 0.375 and 0.5, respectively.

the previous section. In fact, in accordance with Definition 5.1 and Theorem 5.2, the measured hexagonality contributes to a higher score to the rounded rectangle (the shape in Fig. 5.2(b)), and as the shape becomes closer to a hexagon starting from a circle, while it returns a decreasing score for a more circular shape. The smallest $\mathcal{H}(S)$ value for the five shapes presented in Fig. 5.2 is 0.9802, and it is reached for a rectangle in Fig. 5.2(a). On the other side, the shape in Fig. 5.2(e) (i.e., a perfect hexagon) has the largest $\mathcal{H}(S)$ value (among the shapes in Fig. 5.2) equal to 1.0000. This is also consistent with theoretically verified results given that, according to the statements of Theorem 5.2, the shape hexagonality measure $\mathcal{H}(S)$ achieves its maximal value 1 if and only if the shape considered is a hexagon. A plot of computed $\mathcal{H}(S)$ values versus increasing values of parameter ε is provided in Fig. 5.3. The plot starts with $\mathcal{H}(S)$ value 0.9802 for a rectangle (the shape in Fig. 5.2(a)), passes through $0.9845 \approx \dfrac{\sqrt{2} \cdot \pi^{3/2}}{8}$, assigned to disc given in Fig. 5.2(c) (representing as well its local minimal value), and then increases to the largest $\mathcal{H}(S)$ value 1.0000, corresponding to the shape (i.e., a hexagon) given in Fig. 5.2(e). Notice also that all the computed $\mathcal{H}(S)$ values are only the approximative ones, given that only the digital images are used for their computation.

Second experiment: In this experiment, we illustrate the robustness of the shape hexagonality measure $\mathcal{H}(S)$ when increasing amounts of noise are added to a perfect hexagon. For an illustration, five hexagons arranged according to the increasing level of added salt and pepper noise are given in Fig. 5.4, together with their assigned $\mathcal{H}(S)$ values. The results presented are averaged over 100 instances of noised hexagons generated at each level of noise. As expected, the largest $\mathcal{H}(S)$ value over the shapes displayed in Fig. 5.4 is 1.0000, corresponding to noise-free (i.e., perfect) hexagon (the shape in Fig. 5.4(a)), and then decreases as the amount of noise level increases. This is following our expectations and theoretical proven results (Theorem

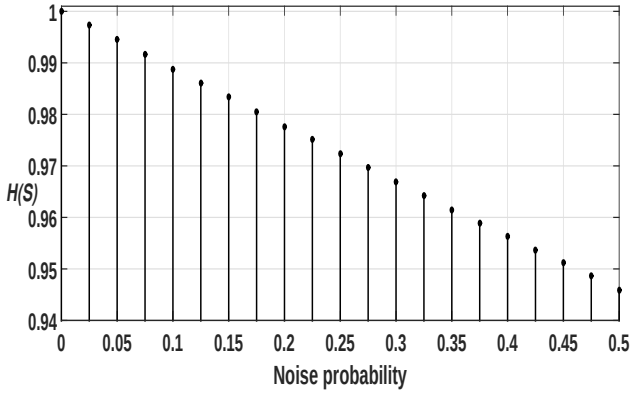


Figure 5.5: Plot of computed $\mathcal{H}(S)$ values for shapes in Fig. 5.4 for increasing amount of salt and pepper noise added to a hexagon given in Fig. 5.4(a).

5.2), since the increased amount of added noise leads to a larger deviation from a perfect hexagon, resulting in the hexagonality return of a decreasing score as a noise amount increases. A plot of computed $\mathcal{H}(S)$ values for values of increasing level of noise is given in Fig. 5.5. The probability of noise ranges from 0, for a perfect hexagon (given in Fig. 5.4(a)) with the largest assigned $\mathcal{H}(S)$ value 1, to 0.5, for a shape in Fig. 5.4(e) with the minimal assigned $\mathcal{H}(S)$ value 0.9459, among the shapes presented.

Third experiment: In the third experiment, we present several more synthetic shapes with aim to illustrate the behaviour of the new hexagonality measure $\mathcal{H}(S)$ in comparison with $Hex_{fit}(S)$. For a fitted hexagon $\mathcal{FH}(S)$, we observe the auxiliary hexagon $Hex(S)$ from (5.9) where the associated parameters $a(S)$ and $c(S)$ are determined from (5.10) and (5.11), or, if necessary, as indicated in Note 5.1. The first six shapes in Fig. 5.6 illustrate how the shape hexagonality measures depend on the presence of holes inside the shape. The shapes presented are obtained from the star shape without holes (the first shape in Fig. 5.6), and differ by the relative size of the holes, their relative position, and also their number inside the shape considered. The shapes are listed in accordance with their decreasing $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ values, which are displayed below each corresponding shape. The hexagonality values measured by $\mathcal{H}_{fit}(S)$ are in the brackets. As expected, as the size of the holes inside the shape increases (e.g., the second and third shapes in Fig. 5.6) then the computed $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ values decrease. The obtained $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ values also show that the hexagonality measures depend on the relative position of the holes inside the shape (the third and fourth shapes, and fifth and sixth shapes in the same figure). Dependence on the number of the holes inside the shape (i.e., star) is illustrated by

0.9392 (0.7031)	0.9215 (0.6545)	0.9005 (0.5745)	0.8503 (0.5451)	0.8342 (0.5011)	0.8131 (0.4047)	0.6615 (0.0079)	0.7341 (0.0063)	0.8195 (0.0033)	0.9071 (0)

Figure 5.6: Several synthetic shapes, together with their assigned $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ values. $\mathcal{H}_{fit}(S)$ values are given in the brackets.

observing the measured $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ values for the fourth and fifth, as well as for the fourth and sixth shapes in Fig. 5.6.

On the other side, the star shape without holes (i.e., the first shape in Fig. 5.6) has the largest measured $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ values, among the shapes presented. These results are also consistent with our perception of how the shape hexagonality should behave, given that a higher hexagonality assigned to the first two shapes in Fig. 5.6 is rather expected than for the fifth and sixth shapes in the same figure. This is also consistent with our initial intention that the hexagonality measures should evaluate to what extent the shape considered differs from a hexagon. Notice also that both hexagonality measures provide the same arrangement of the shapes, which can be beneficial in some applications.

The last four shapes in Fig. 5.6 demonstrate how the two hexagonality measures might establish a different ranking among the shapes considered. The shapes presented, listed with respect to their increasing $\mathcal{H}(S)$ values, represent the shape of a circle line (the seventh shape in Fig. 5.6), whereas the other three shapes are obtained by successive excluding the quarters of the circle line. If the same shapes are arranged in accordance with their increasing $\mathcal{H}_{fit}(S)$ values then the opposite order is obtained. Indeed, a monotonic increase of the measured $\mathcal{H}(S)$ hexagonality corresponds to a monotonic decrease of the hexagonality measured by $\mathcal{H}_{fit}(S)$. Herein the changes in the measured hexagonality $\mathcal{H}(S)$ do support our expectations, which can be considered an advantage over $\mathcal{H}_{fit}(S)$. This comes from the fact that most hexagonal shape (among these four shapes) is the last shape in Fig. 5.6 with measured $\mathcal{H}(S)$ value equal to 0.9071, while the least hexagonal is the seventh shape in the same figure (i.e., a circle line) with measured $\mathcal{H}(S)$ hexagonality equals 0.6615. Also, it is worth mentioning that the lowest measured $\mathcal{H}_{fit}(S)$ equals 0 is assigned to the last shape in Fig. 5.6, which can also be understood as its disadvantage over a new hexagonality measure $\mathcal{H}(S)$. Besides, different rankings among the shapes presented can also be recognized as a desirable in some applications, given that in such cases these two hexagonality measures can be combined and used together to improve performance of various object analysis tools. This latter will be particularly

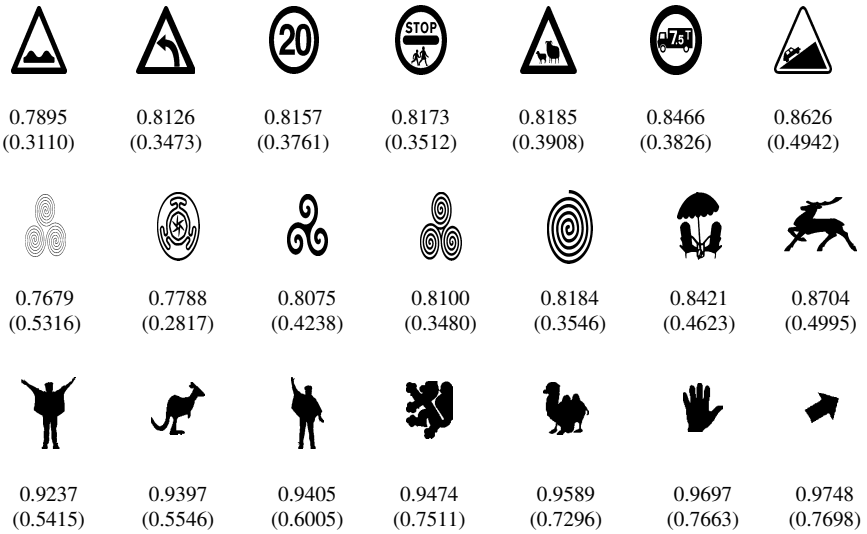


Figure 5.7: Randomly selected shapes, and their computed hexagonalities $\mathcal{H}(S)$ (given below each shape). The computed $\mathcal{H}_{fit}(S)$ values are in the brackets.

demonstrated in the experiments in Section 5.5.

5.3.2 Illustrating behaviour of $\mathcal{H}(S)$ on real image data

Herein we provide several experiments to illustrate the behaviour of the new hexagonality measure $\mathcal{H}(S)$ on real image data. To demonstrate how new measure matches human perception, it is applied to diverse shapes from several image datasets already discussed by others in various analysis tasks. The examples provided in the last experiment illustrate the new hexagonality behaviour under various shape deformations.

Fourth experiment: This experiment illustrates the behaviour of the new hexagonality measure $\mathcal{H}(S)$ applied to various randomly selected shapes. Figure 5.7 presents several such shapes together with their assigned $\mathcal{H}(S)$ values, while the measured $\mathcal{H}_{fit}(S)$ values are given in the brackets. In the first row in Fig. 5.7, the seven shapes of traffic signs are ranked in accordance with their increasing $\mathcal{H}(S)$ values. If the same shapes are listed with respect to $\mathcal{H}_{fit}(S)$ hexagonality, a slightly different ranking is obtained. In fact, the third and fourth, as well as the fifth and sixth traffic sign shapes should change their places if the ranking according to increasing $\mathcal{H}_{fit}(S)$ is applied. It can be said that such obtained ranking is not in accordance with our perception of the behaviour of shape hexagonality measure since we expect a higher

hexagonality assigned to the fourth shape in the first row in Fig. 5.7 rather than for the third shape in the same row. Also, a higher hexagonality assigned to the sixth shape is more expectable than for the fifth shape in the same row.

Several more shapes are presented in the remaining two rows in Fig. 5.7. The shapes presented are also arranged according to their assigned $\mathcal{H}(S)$ values. It can be said that the changes in the measured hexagonality follow the changes in the overall structure of the shapes considered. Although some of the shapes presented can differ a lot, there exist also those that are similar in their nature (for example, the first and fourth shapes in the second row in Fig. 5.7, as well as the first and third shapes in the third row). Notice that the order of the first and fourth shapes in the second row is changed if $\mathcal{H}_{fit}(S)$ is applied. Such a property of $\mathcal{H}_{fit}(S)$ can be considered as its drawback compared to $\mathcal{H}(S)$. On the other side, the differences in the measured $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ values that have been assigned to the human shapes (the first and third shapes in the third row) do not contradict our perception about the behaviour of the shape hexagonality measures. In fact, we prefer a higher hexagonality assigned to the human shape with arms spread parallel to the body (the third shape in the last row) than for the shape with arms spread diagonally (the first shape in the same row). The obtained results are also consistent with our initial requirement that the hexagonality measures should quantify how much a given shape differs from the corresponding hexagon. Notice also that the rankings with respect to both hexagonality measures, among the shapes in the last row, differ only for the fourth and fifth shapes. Such a property of $\mathcal{H}_{fit}(S)$ can be viewed as its weakness over $\mathcal{H}(S)$ since the fifth shape is recognized to be more hexagonal than the fourth shape in the same row. Finally, it can be noticed that the shapes with higher assigned hexagonality, listed in the third row in Fig. 5.7, represent more compact shapes containing the larger portions of the object pixels, among all the shapes in Fig. 5.7. These results are in accordance with human perception, since as most compact shape, among the shapes presented, can be understood the shape of the arrow (the last shape in the third row) with the largest measured $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ hexagonality equal to 0.9748 and 0.7698, respectively.

To further demonstrate the properties of the new hexagonality measure, it is applied to two well-known image datasets. The first one is the animal dataset [10] containing 20 species of animals, with each having 100 images, i.e., a total of 2000 animal images. The shapes considered have been obtained from real images through a variation in pose, viewing angle, articulation, self-occlusion, etc. The ten lowest and ten highest arranged shapes with respect to their increasing $\mathcal{H}(S)$ values are given in Fig. 5.8, together with ten middle arranged shapes. Such an established arrangement corresponds to our perception, given that $\mathcal{H}(S)$, according to Definition 5.1, should measure how much a given shape differs from its corresponding hexagon $Hex(S)$ (given in (5.9)), independently of how much the ratios of the corresponding semi-axes differ. This further explains: **1)** why $\mathcal{H}(S)$ returns lower values for the shapes in the first row, characterized by increased presence of the long thin and/or

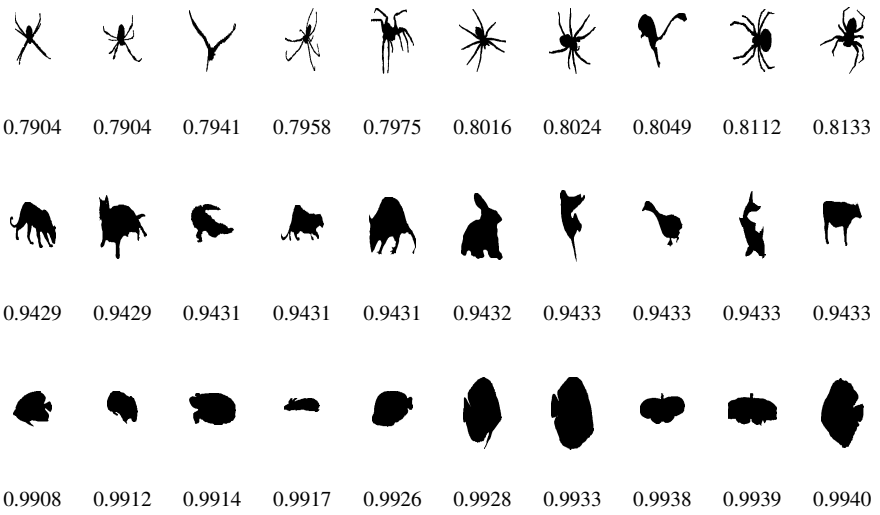


Figure 5.8: The ten lowest (first row), intermediate (second row), and highest (third row) ranked shapes according to increasing $\mathcal{H}(S)$ values from animal database [10]. Computed $\mathcal{H}(S)$ values are given below an appropriate shape.

elongated protrusions (e.g., spider's legs, mouse's tail, etc.), whereas increased $\mathcal{H}(S)$ values are assigned to more compact shapes containing larger portions of the object's pixels (the shapes in the second and third row), and **2)** why some shapes are assigned similar hexagonality even though they correspond to different animal species (e.g., the shapes in the second row, the ninth and tenth shapes in the third row, etc.).

The second well-known dataset is MPEG-7 CE-1 [73], containing a total of 1400 shapes (70 different classes, each having 20 shapes). The ten highest, intermediate and lowest ranked shapes with respect to new $\mathcal{H}(S)$ measure are in Fig. 5.9. The measured $\mathcal{H}(S)$ values are given below each corresponding shape. It can be said that shapes which contain long thin or elongated portions, or which are sparse with larger areas of non-object pixels, are recognized by $\mathcal{H}(S)$ as less hexagonal shapes. On the other side, although there is no ideal hexagon in the observed dataset, it can be said that the ten highest ranked shapes (i.e., the shapes of pencils and squares with different level of shape deformations) can be viewed as most hexagonal shapes, among the shapes presented in Fig. 5.9. This is also in accordance with the theoretical considerations, since a square can be understood as a special case of a hexagon as well (for details, see formulas in (5.6) and (5.7)). As expected, the largest measured hexagonality $\mathcal{H}(S)$ is assigned to the shape of the pencil sharpened on both sides,

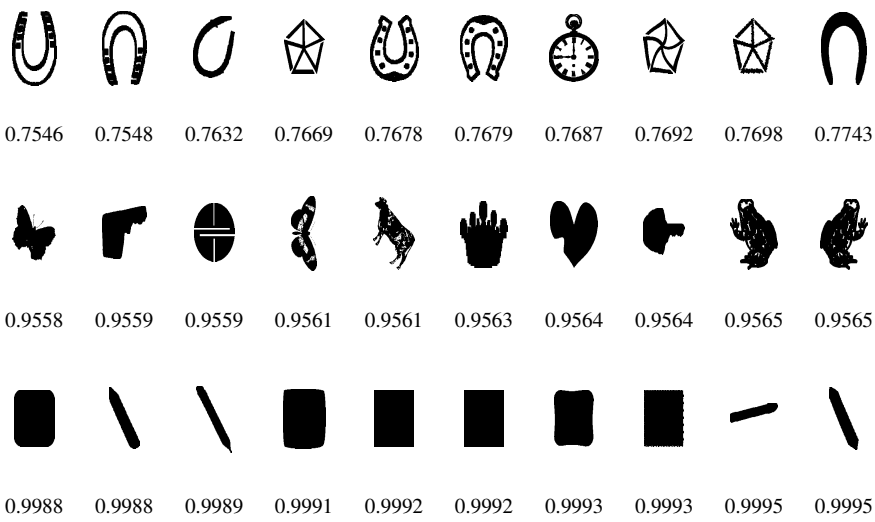


Figure 5.9: The ten lowest (first row), intermediate (second row) and highest (third row) arranged shapes according to $\mathcal{H}(S)$ hexagonality from MPEG-7 CE-1 dataset [73].

representing simultaneously most hexagonal shape among the shapes discussed. It is worth noting that there are several shapes having a similar assigned hexagonality. For example, the first and second shapes of the horseshoes in the first row, as well as the third and fourth shapes of the horseshoes in the same row, or the shapes of the frogs in the second row, etc. These results are also in accordance with the results proven in Theorem 5.2 that the new hexagonality measure $\mathcal{H}(S)$ is also invariant to similarity transformations.

Fifth experiment: In this experiment, we illustrate how the shape deformations contribute to changes in the measured hexagonality $\mathcal{H}(S)$. To demonstrate this, several shapes, extracted from a time series (e.g., video), are given in Fig. 5.10, together with their assigned $\mathcal{H}(S)$ values. The shapes presented are arranged according to the time in which they have been extracted from the time series. Thus, for instance, the first nine shapes in Fig. 5.10, representing a human gait, correspond to different appearances of human silhouettes in a sequence of nine consecutive frames. It can be noticed that the corresponding shape deformations lead to the differences in the measured $\mathcal{H}(S)$ hexagonality. These changes correspond to our perception of how $\mathcal{H}(S)$ behaves, following from the theoretical and empirical results that have been already discussed and illustrated. For example, if the legs of the observed walker are closer to

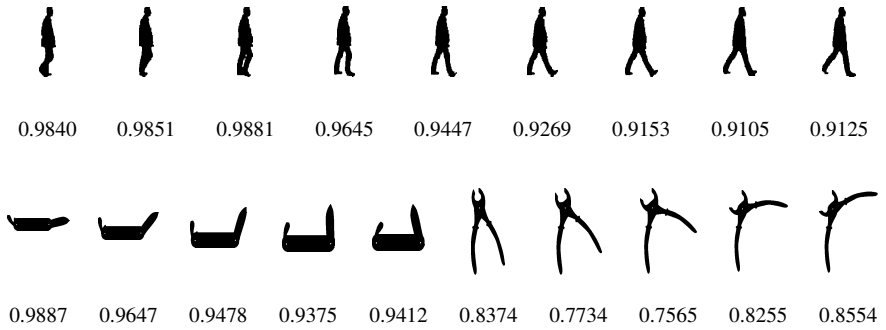


Figure 5.10: Examples illustrating the behaviour of $\mathcal{H}(S)$ under various shape deformations. $\mathcal{H}(S)$ values are given below the corresponding shape. The first row represents a human gait, while the second row illustrates closing of pocketknife (the first five shapes) and opening tongs (the latter five shapes).

each other, then lesser area of the white pixels between them contributes to increasing the hexagonality $\mathcal{H}(S)$ assigned to them. Because of this, we could expect that the shapes from the first half of the time sequence in the first row in Fig. 5.10 have higher hexagonality measured by $\mathcal{H}(S)$ than the shapes extracted from the second half of the same time sequence. These results are also in accordance with theoretical observations, given that, according to Definition 5.1, a new hexagonality measure $\mathcal{H}(S)$ evaluates a degree of similarity of the measured shape to its corresponding hexagon $Hex(S)$, as defined in (5.9). The same argumentation applies to the series of shapes presented in the second row. The first five shapes illustrate the deformations of the pocketknife shape during its closing, whereas the second five shapes represent how the shape of tongs changes when it is opening. The results obtained again confirm our expectations, as well as the theoretical observations. It can also be noticed that the shapes of tongs have the lower measured hexagonality $\mathcal{H}(S)$, given that they can be considered to be less hexagonal than the remaining shapes in Fig. 5.10. This latter is also supported by the fact that the shapes of tongs contain larger proportions of non-object pixels in relation to all remaining shapes in Fig. 5.10.

5.4 Side results of the new hexagonality measure $\mathcal{H}$(S)

The main result of this chapter, i.e., a new hexagonality measure $\mathcal{H}(S)$, provides several important consequences. The following two subsections shall describe them in a more formal way, and list some desirable properties they satisfy.

5.4.1 Measuring orientation based on $\mathcal{H}$(S)

In this subsection, we illustrate a new approach to determining the shape orientation obtained as a by-product of the method for computing a new hexagonality measure. In fact, following the formula in (5.21), we can conclude that the new hexagonality measure essentially represents the quantity optimized over all the rotations S_θ of the shape S around its centroid, while θ runs through $[0, 2\pi)$. Regarding this, it is natural to define the orientation of a given shape S as an angle (i.e., orientation) minimizing the integral in (5.21) for all orientations from $[0, 2\pi)$. In the following, this orientation will be referred to as *hexagonality-based shape orientation*, and it is formally defined as follows.

Definition 5.2 *The orientation of a given shape S is defined by an angle θ for which the integral in* (5.21) *achieves the minimal value.*

Several shapes to illustrate the properties of such a defined method for computing shape orientation are given in Fig. 5.11. The orientations computed by standard method [152] are in the brackets. A few animal shapes taken from the animal dataset [10] (already mentioned in the fourth experiment) are shown in the first two rows in Fig. 5.11. The shapes presented are ranked such that the shapes recognized to be more compact are in the first row, while animal shapes that can be considered as more elongated are given in the second row. It can be said that the computed orientations are in accordance with our perception, given that measured orientations also coincide with a certain axis of the shape elongation. In addition, it is worth noting that orientations determined by these two competitive methods are very similar or even coincide. This latter comes from the fact that the measured orientations do not depend only on the nature of the object itself, but also on the fact that the original shapes are replaced by their digital counterparts. This further implies that computed orientations are given as the approximative ones. The similar reasoning applies to the shapes of the automotive's brands from [1] that are given in the last two rows in Fig. 5.11. It can be noticed that all the orientations are also consistent with human perception of what can be expected to be a shape orientation. However, among the shapes presented, the fourth and sixth shapes in the third row in Fig. 5.11 deserve special attention, given that these two shapes have three axes of symmetry, and they are 3-fold rotationally symmetric shapes. For this reason, it is difficult to say what their orientations should be. This is something that we expected, given that there are three potential modulo 180° orientations which can be equally selected as a natural choice for shape orientation. Computed orientations are closely aligned with the corresponding axes of the object symmetry, implying that these two competitive methods are actually complementary and mostly compatible. This behaviour of the new method can be considered desirable, given that, due to variations in shapes and diversity of applications, there is no single method for computing the orientation that can be efficiently and also successfully applied to all shapes.

5.4.2 New elongation measure

The quantities $a(S)$ and $c(S)$ associated with a given shape S, as defined in (5.10) and (5.11), or, if necessary, as stated in Note 5.1, can also be employed to measure a shape elongation. Herein such a derived measure will be denoted as $\mathcal{E}(S)$, and its formal definition is given as follows.

Definition 5.3 *Let a shape S, having the centroid coincident with the origin, be given. The elongation measure $\mathcal{E}(S)$ of a given shape S is defined as a ratio of the lengths of the longer and shorter semi-axis of the corresponding auxiliary hexagon $Hex(S)$, as defined in* (5.9)*, i.e.,*

$$\mathcal{E}(S) = \frac{a(S)}{a(S) - c(S)}. \tag{5.32}$$

Several desirable properties of the new elongation $\mathcal{E}(S)$, expected to be satisfied by all elongation measures, are summarized in the following theorem.

Theorem 5.3 *Given a shape S whose centroid coincides with the origin. The new elongation measure $\mathcal{E}(S)$ has the following properties:*

(a) $\mathcal{E}(S) \in [1, \infty)$, for all shapes S;

(b) $\mathcal{E}(S) = 1 \Leftrightarrow$ *Hex(S)* is a degenerative hexagon (i.e., a square);

(c) $\mathcal{E}(S)$ is invariant with respect to translation, rotation and scaling transformations.

Proof 5.3 *Items (a) and (b) follow directly from the definition of $\mathcal{E}(S)$, and properties of the quantities $a(S)$ and $c(S)$. Indeed, in accordance with* (5.10) *and* (5.11)*, and also, if necessary, in accordance with Note 5.1, it follows that $a(S) > c(S) \geq 0$, and, consequently,*

$$\mathcal{E}(S) \;=\; \frac{a(S)}{a(S) - c(S)} \;=\; 1 + \frac{c(S)}{a(S) - c(S)} \;\geq\; 1, \tag{5.33}$$

for all shapes S. Now, it is enough to consider an isothetic hexagon $Hex(t)$ for which the corresponding parameters $a(t)$ and $c(t)$ are equal to t and $t-1$, for $t \geq 1$. Based on this, it is easy to show both $\mathcal{E}(Hex(t)) = t$, and also while t ranges through $[1, \infty)$ then $\mathcal{E}(Hex(t))$ reaches all values from $[1, \infty)$. Moreover, we can conclude that the minimal possible value equals 1 is reached for $t = 1$, i.e., when $c(t) = 0$, implying further that corresponding hexagon $Hex(t)$ is then a square (i.e., degenerative hexagon). The item (c) follows from the fact that $\mathcal{E}(S)$ is computed from $a(S)$ and $c(S)$, which

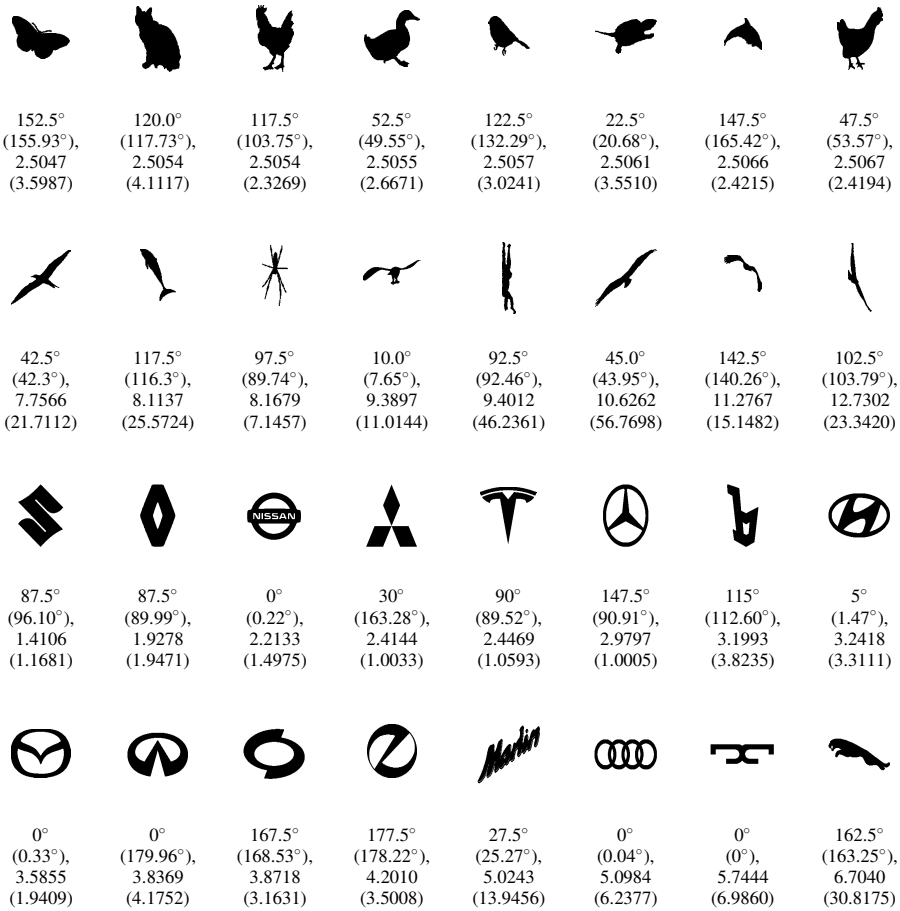


Figure 5.11: Several randomly selected shapes together with their computed orientations (the first and second angle are given immediately below the shape), and their measured elongations (the third and fourth assigned number below the corresponding shape). Orientations as well as elongations of the shapes computed by the standard methods are given in the brackets.

are translational and rotational invariants by definition, whereas (5.23) *and* (5.24) *give*

$$\mathcal{E}(\lambda \cdot S) \;=\; \frac{a(\lambda \cdot S)}{a(\lambda \cdot S) - c(\lambda \cdot S)} \;=\; \frac{a(S)}{a(S) - c(S)} \;=\; \mathcal{E}(S), \tag{5.34}$$

i.e., $\mathcal{E}(S)$ is a scaling invariant. This completes the proof of the theorem. □

Examples in Fig. 5.11 can also be used to illustrate how $\mathcal{E}(S)$ acts. The computed values of the new $\mathcal{E}(S)$ and the standard $\mathcal{E}_{st}(S)$ elongation measure from[152] are given as the third and fourth number below each corresponding shape. All the measured $\mathcal{E}(S)$ and $\mathcal{E}_{st}(S)$ values are given as the approximative, following that digital images are used for their computations rather than their original counterparts. It can be said that measured $\mathcal{E}(S)$ values match our expectations, given that we expect that the shapes which are elongated (e.g., the shapes presented in the second row) have a higher assigned elongation than the shapes which are more compact (e.g., the shapes in the first row in the same figure). The similar discussion applies to shapes of the automotive logos from [1] presented in the following two rows in Fig. 5.11. Notice also that measured $\mathcal{E}_{st}(S)$ values for the fourth and sixth shapes of the logos are approximately equal to 1. This is also consistent with something we expected, given that these shapes are 3-fold rotationally symmetric for which, as already mentioned, the standard method does not work as expected. Indeed, in accordance with the theoretical arguments relating to the standard elongation measure $\mathcal{E}_{st}(S)$ (e.g., see Lemma 1 in [185]), we have that measured standard elongation $\mathcal{E}_{st}(S)$ for such shapes is approximately equal to 1. Contrary to this, the new elongation measure is able to provide a clear distinction among these shapes, which can be considered to be its essential advantage compared to $\mathcal{E}_{st}(S)$. Notice as well that shapes of the animals and automotive brands are arranged according to increasing $\mathcal{E}(S)$ values. In the following, we will present few more experiments in order to illustrate the applicability and usability of $\mathcal{E}(S)$ in various object classification tasks.

5.5 Effectiveness of hexagonality in different applications

In this section, we illustrate how the new shape hexagonality measure $\mathcal{H}(S)$ can be employed efficiently in diverse classification tasks, together with hexagonality $\mathcal{H}_{fit}(S)$ and new elongation measure $\mathcal{E}(S)$. Given that these new shape measures are not designed for a specific application, we present herein a number of different experiments aimed to illustrate advantages as well as the importance of the results obtained by their use, instead of applying already known, possible complex tools which are known to provide improvement in such tasks. For instance, in some experiments, emphasis will not be on the best possible score for the observed database, but rather on implementing only a few simple and fast to compute shape descriptors. To illustrate the generality and effectiveness of the new shape-based measures, we present several classification experiments performed on a number of well-known image databases.

Sixth experiment: First classification task. In this experiment we illustrate that new, introduced in this chapter, shape measures can be combined, and also used

together with other already known shape descriptors to improve their performance when classifying images taken from MPEG-7 CE-1 image database [73]. As already mentioned, this image database consists of 70 different classes, each having 20 shape images. Several shape examples belonging to this image dataset are shown in Fig. 5.9. It is worth mentioning that this classification task has become very challenging in recent years from the research point of view, but also from the aspect of application in a wide spectrum of object analysis tasks. For the classification, we have used the nearest neighbor classifier with a Mahalanobis distance. The experiments presented are divided into four groups depending on the initial set of selected shape descriptors. The results achieved using only several shape descriptors, or when they are combined with new shape measures are shown in Table 5.1.

In the first group of experiments, a set of five global descriptors such as rectangularity, roundness, compactness, area-based and perimeter-based convexities[2] along with the multi-component shape measure $\mathcal{D}(S)$ [162] achieves a leave-one-out classification accuracy of 78.50%. If the new $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ are added into consideration separately, the classification rate has increased to 82.50% and 83.50%, respectively. Combining $\mathcal{H}_{fit}(S)$ and $\mathcal{E}(S)$ together with the initial set of descriptors, the classification result has further improved to 85.36%. Notice that such a good result provides relatively high improvement in accuracy of nearly 7%, which is significantly better than the relative improvement of approximately 3.7% achieved in [122] when the squareness measures are added to the corresponding descriptors selected by the authors. The last three experiments further show that new shape measures are complementary, and can be used together to increase efficiency in this classification task. This can be illustrated when only $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ are combined together or separately with $\mathcal{E}(S)$, without initial set of descriptors. For example, combining only $\mathcal{H}_{fit}(S)$ and $\mathcal{E}(S)$ gives accuracy of 49.07%, whereas together with $\mathcal{H}(S)$ the classification efficiency improves to 61.86%.

In the following group of experiments, the area-based anisotropy $\mathcal{A}(S)$ [115] was used instead of $\mathcal{D}(S)$. Such combined shape descriptors provide efficiency of 76.93%. If the new elongation measure $\mathcal{E}(S)$ is added, then the achieved efficiency improves to 79.50%, whereas by adding the new hexagonality measure $\mathcal{H}(S)$ instead, a better accuracy of 82.00% is provided. Including the hexagonality $\mathcal{H}(S)$ combined separately with $\mathcal{E}(S)$ and $\mathcal{H}_{fit}(S)$, the classification accuracy has further increased to 83.57% and 84.79%, respectively. A better result is obtained if the two hexagonality measures are used together with new elongation measure. The classification rate is then increased to 85.50%. Such a big improvement in accuracy of approximately

[2]The five global applied shape descriptors are *rectangularity* based on the ratio of the areas of shape S and its minimum bounding rectangle [119], *roundness* based on the ratio of $Area(S \cap C_{fit})$ and $Area(S)$ where C_{fit} represents the corresponding fitted circle with the same centroid, and equal area as the shape S [96], *compactness* given as the ratio of the squared perimeter and area multiplied by $4 \cdot \pi$ [152], *area-based convexity* defined as the ratio of the areas of the shape S and its convex hull and *perimeter-based convexity* given as the ratio of the perimeters of the shape S and its convex hull [159].

Table 5.1: Classification rates obtained for different choices of the initial shape descriptors augmented by new shape measures. Improvements in accuracy obtained including the new shape measures are bolded.

	Shape features	Rates
First group	6 shape features	78.50%
	6 shape features + $\mathcal{H}(S)$	**82.50**%
	6 shape features + $\mathcal{H}_{fit}(S)$	**83.50**%
	6 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{E}(S)$	**85.36**%
Second group	6 shape features	76.93%
	6 shape features + $\mathcal{E}(S)$	**79.50**%
	6 shape features + $\mathcal{H}(S)$	**82.00**%
	6 shape features + $\mathcal{H}(S)$ + $\mathcal{E}(S)$	**83.57**%
	6 shape features + $\mathcal{H}(S)$ + $\mathcal{H}_{fit}(S)$	**84.79**%
	6 shape features + $\mathcal{H}(S)$ + $\mathcal{H}_{fit}(S)$ + $\mathcal{E}(S)$	**85.50**%
Third group	7 shape features	79.79%
	7 shape features + $\mathcal{E}(S)$	**81.71**%
	7 shape features + $\mathcal{H}(S)$	**83.14**%
	7 shape features + $\mathcal{H}_{fit}(S)$	**83.57**%
	7 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{E}(S)$	**85.71**%
Fourth group	8 shape features	83.00%
	8 shape features + $\mathcal{H}(S)$	**84.50**%
	8 shape features + $\mathcal{E}(S)$	**84.86**%
	8 shape features + $\mathcal{H}_{fit}(S)$	**86.71**%
	8 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{H}(S)$	**87.86**%
	8 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{E}(S)$	**87.93**%

8.6% also illustrates the importance and usefulness of the new shape measures, as mentioned in the previous group of experiments. It can also be concluded that two hexagonality measures, together with new elongation measure, supply mostly independent shape information, keeping the classification accuracy at a relatively high level.

The third group of experiments represents some kind of mixture of the first two groups, where the initial set of the global descriptors[2] is further strengthened with anisotropy $\mathcal{A}(S)$ and disconnectedness $\mathcal{D}(S)$. These seven shape descriptors provide a classification efficiency of 79.79%. When the new elongation $\mathcal{E}(S)$, as well as the two hexagonality measures $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ has been added separately, the reached classification has improved to 81.71%, 83.14% and 83.57%, respectively. If the hexagonality measure $\mathcal{H}_{fit}(S)$ is further augmented with new elongation $\mathcal{E}(S)$ measure, the classification efficiency has increased to 85.71%. This pretty good result also represents a very high relative improvement of nearly 6% over the initial set of shape descriptors described in this group of experiments. It can also be noticed that such a high efficiency also outperforms the highest accuracy in the previous group of experiments using the same number of shape descriptors (i.e., a total of 9 descriptors).

In the fourth experimental group, we observe an improvement in efficiency that is reached by implementing new introduced shape measures, relative to already mentioned five global shape descriptors[2] augmented by the ellipticity measure (denoted as $\mathcal{E}_P(S)$ in [160]), as well as the second and third affine moment invariants [57]. The classification score using these shape features is 83.00%. When the new hexagonality $\mathcal{H}(S)$ and new elongation $\mathcal{E}(S)$ measures were added separately, the classification rate has increased to 84.50% and 84.86%, respectively. A better accuracy is provided by adding the hexagonality measure $\mathcal{H}_{fit}(S)$ instead: 86.71%, while a further improvement of 87.86% was obtained by combining two hexagonality measures $\mathcal{H}_{fit}(S)$ and $\mathcal{H}(S)$ along with set of the initial shape descriptors. The highest improvement of 87.93% was obtained if the hexagonality $\mathcal{H}_{fit}(S)$ measure is used together with new elongation $\mathcal{E}(S)$ measure. Notice that efficiency of 87.93% represents the best achieved classification rate (i.e., a benchmark result), whereas the rate of 87.86% represents the second highest classification result obtained for this image dataset.

In order to further point out the importance and benefits of the new shape measures, we compare our results with rates obtained by several authors until now. For example, the best-achieved result on this dataset until recently has been 87.13% [4] which was obtained using the topology of the objects and curvature tree to model the shape. On the other side, using the new, in this chapter introduced, shape measures together with several other simple, and also fast to compute shape descriptors, the two highest accuracies of 87.86% and 87.93% outperform the efficiency achieved in [4], where the highest achieved accuracy of 87.93% also represents the benchmark result on this image dataset. Another example that deserves attention is a score of 85.40%, obtained using the inner distance shape context [90]. Moreover, as it can be seen from Table 5.1, there exist several combinations of the shape features that are complementary, and which can be combined to outperform this efficiency (for instance, the last experiments in the second and third group of experiments, or the last three experiments in the fourth group of experiments). Finally, we provide several more

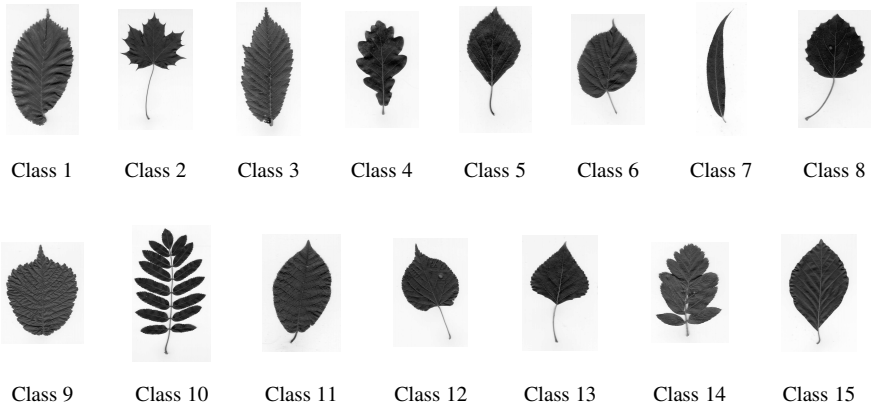


Figure 5.12: Shape samples of leaf images from the Swedish leaf dataset [151]. One sample shape per each leaf specie is presented.

results that have been published so far (relating to the same image dataset), aiming at further illustration of the quality and usefulness of the new shape measures. To mention a few: curvature scale space (CSS) [101] 75.44%, shape context (SC) [13] 76.51%, curve edit distance [131] 78.17%, pattern spectrum and local binary pattern (PS+LBP) [137] 79.38%, skeletal shape context (SSC) [171] 79.92%, generative model [156] 80.03%, etc.

Seventh experiment: Second classification task. This experiment illustrates the effectiveness of the new hexagonality measures $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ together with new elongation measure $\mathcal{E}(S)$ in the classification of leaf images taken from Swedish Leaf dataset [151]. This dataset contains a total of 1125 colored leaf images, classified into 15 different species, with 75 images per each species. Examples of leaf images for each species are shown in Fig. 5.12. It should be noted that there exist very high inter-species similarities among the presented examples, which makes the classification task performed on this database a very demanding one. Instead, Rhouma et al. [115] have studied the dataset of 675 images consisting of 9 different species, each having 75 images. Even though the classification task was performed on a shortened (i.e., with reduced inter-species similarities) image dataset, the efficiency of 97% obtained using seven multi-component shape invariants [115] and seven Hu moment invariants [66] (i.e., a total of 14 shape features) represents essentially a very good result.

In our experimental study, we observe two groups of experiments, designed according to the initial set of the shape descriptors used in the experiment. The classification accuracies achieved using the nearest neighbor classifier with a Mahalanobis distance are in Table 5.2. We start with rectangularity, area-based convexity[2] and

Table 5.2: Rates obtained combining different shape features together with new descriptors. Results achieved including new shape measures are bolded.

	Shape features	Rates
First group	5 shape features	94.96%
	5 shape features + $\mathcal{H}_{fit}(S)$	**96.15**%
	5 shape features + $\mathcal{E}(S)$	**96.59**%
	5 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{E}(S)$	**97.63**%
	5 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{H}(S)$	**97.93**%
	5 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{H}(S)$ + $\mathcal{E}(S)$	**98.07**%
Second group	6 shape features	96.15%
	6 shape features + $\mathcal{E}(S)$	**96.89**%
	6 shape features + $\mathcal{H}_{fit}(S)$	**97.63**%
	6 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{E}(S)$	**97.93**%
	6 shape features + $\mathcal{H}_{fit}(S)$ + $\mathcal{H}(S)$	**98.67**%

multi-component shape disconnectedness measure $\mathcal{D}(S)$ [162]. In the first group of experiments, the initial set of descriptors is further extended with roundness^2 and multi-component shape measure from [115] (denoted as $IImca(S)$). Such a defined set of five simple descriptors reaches a leave-one-out classification rate of 94.96%. Including the hexagonality measure $\mathcal{H}_{fit}(S)$ and new elongation measure $\mathcal{E}(S)$ separately, the classification rate has increased to 96.15% and 96.59%, respectively. Combining both $\mathcal{H}_{fit}(S)$ and $\mathcal{E}(S)$, a better accuracy of 97.63% is achieved. If the hexagonality measure $\mathcal{H}_{fit}(S)$ is further strengthened with new hexagonality measure $\mathcal{H}(S)$, and additionally with both $\mathcal{H}(S)$ and $\mathcal{E}(S)$, the classification efficiency has increased to 97.93% and 98.07%, respectively. This relatively high improvement of nearly 3.1% represents a very good classification result outperforming the best-achieved classification rate of 97% obtained in [115], given that only eight simple shape descriptors have been used.

In the second group of experiments, the initial set of descriptors is augmented with the first Hu moment invariant [66], perimeter-based convexity^2 and multi-component shape measure from [115] (denoted as $IImcc(S)$). This set of descriptors achieves a leave-one-out classification accuracy of 96.15%. If the new elongation measure $\mathcal{E}(S)$ and hexagonality $\mathcal{H}_{fit}(S)$ are included separately, the efficiency has increased to 96.89% and 97.63%, respectively. Combining both $\mathcal{E}(S)$ and $\mathcal{H}_{fit}(S)$ together with the initial descriptors, the classification rate has increased to 97.93%. A further

improvement of 98.67% has been achieved if the initial set of descriptors is strengthened with $\mathcal{H}_{fit}(S)$ and $\mathcal{H}(S)$ instead. It should be noticed that this relatively high improvement represents a better result than those achieved in the first group of experiments, and, consequently, better than the best-achieved in [115] for a total of 8 shape descriptors. Such an obtained efficiency of nearly 99% also represents the best-achieved result (i.e., a benchmark result) performed on this dataset.

It can be said that all the experiments performed illustrate that new hexagonality measures $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$, and new elongation measure $\mathcal{E}(S)$ are relatively efficient for such a classification task. As can be seen from Table 5.2, all these relatively high accuracies, obtained using not more than eight simple shape features, outperform the efficiency of 89.1% (as stated in [162]) using only seven Hu moment invariants [66] as well as the efficiency of 94.9% obtained using seven multi-component moment invariants introduced in [115]. This can also be illustrated observing only the new shape measures, without others in these experiments used initial shape descriptors. For example, when we perform classification using only the hexagonality measure $\mathcal{H}_{fit}(S)$, we have achieved an accuracy of 53.04%. Better accuracy was provided by including separately $\mathcal{H}(S)$ and $\mathcal{E}(S)$: 54.07% and 63.26%, respectively. Combining two hexagonality measures we achieved an efficiency of 75.56%, whereas $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ separately with elongation measure $\mathcal{E}(S)$ improve accuracy to 84.00% and 84.59%, respectively. A further improvement of 87.85% was obtained by combining all three new shape measures.

Although our experiments were performed on a shortened image database of only 9 classes, these experiments suggest that combining new shape measures can be beneficial in the tasks of leaf classification. The obtained classification scores (using only a small number of shape features and simpler classifier) are comparable and pretty competitive to those object-based methods, specifically designed for working on this image database. Usually, such methods involve a much larger set of color and texture leaf features, together with complex classifier. Some of them are: classification rate of 82.40% achieved by Söderkvist in [151] using 20 object features; scores of 88.12% and 89.60% obtained using the shape context and dynamic programming (SC+DP) [90] and Fourier descriptors (FDs) [90], respectively, with more than 128 used features; 94.67 using the distance transform network (DNT) and SVM classifier, with more than 100 object features [116]; then rate of 95.33% obtained using the shortest path texture context and dynamic programming (SPTC+DP) with more than 128 object features [90]; and rates of 96.28% and 96.53% using the shape tree (ST) [54] and multiscale triangle two side lengths and angle representation (TSLA) [104] with more than 100 and 120 applied object features, respectively.

Eighth experiment: Third classification task. In this experiment, we show that new hexagonality measures can be successfully employed together with new elongation measure in the classification task performed on the database of color galaxy images taken from the Galaxy Zoo dataset [92]. This well-known image database

Figure 5.13: Examples of the elliptical, spiral and edge-on galaxies selected from the Galaxy Zoo database [92], together with their extracted shapes.

Table 5.3: Results obtained using only five simple features along with new shape measures. Scores achieved including new shape measures are given bolded.

Shape features	Rates
5 shape features	77.44%
5 shape features + $\mathcal{E}(S)$	**80.31**%
5 shape features + $\mathcal{H}(S)$	**81.26**%
5 shape features + $\mathcal{H}(S)$ + $\mathcal{E}(S)$	**81.45**%
5 shape features + $\mathcal{H}_{fit}(S)$	**84.32**%

was previously studied by Shamir [135, 136] to design algorithms for automatic analysis and classification of galaxy images as the elliptical, spiral and edge-on galaxies. Examples of images of galaxies together with their shapes obtained through using the Otsu algorithm [107] are displayed in Fig. 5.13. In [136] the author described the algorithm, called Ganalyzer, achieving the efficiency of 88.76% with the gold standard adopted on the base of a manual classification performed by the author. As such, it is not so reliable due to existence of both galaxy images which are in-between classes and features of the galaxies that are difficult to notice using the human eye.

In our experiment, we demonstrate a simpler approach based on the use of a few simple already mentioned shape descriptors together with our new shape measures in order to illustrate how the simple shape features can be reasonably efficient in this classification task. The classification scores are shown in Table 5.3. The initial set of shape features, used in this experiment, consists of roundness^2, the first and fifth moment invariant by Hu [66], the third affine moment invariant [57] and multi-component shape measure $Imcc(S)$ from [115]. The nearest neighbor classifier based on a Mahalanobis distance achieves leave-one-out classification score of 77.44%.

The classification rate has increased to 80.31% and 81.26% by including separately the new elongation measure $\mathcal{E}(S)$ and new hexagonality measure $\mathcal{H}(S)$, respectively. A better result of 81.45% was obtained if the initial set was strengthened by both $\mathcal{H}(S)$ and $\mathcal{E}(S)$. A further improvement of 84.32% was achieved by including only the hexagonality measure $\mathcal{H}_{fit}(S)$. This very good classification rate, achieved using only six simple shape features, illustrates also a very high relative improvement of nearly 7% as a result of including new, in this chapter derived, shape measures. Besides, it can be said that this result is also comparable to the rate of 88.76% obtained in [136], as well as to the recent score of 88.55% achieved in [162]. Also, we noticed that new shape measures (separately and/or together) can be relatively effective in this classification task. In fact, if we performed classification using only the hexagonality measure $\mathcal{H}(S)$, then the achieved classification rate was 53.15%. This result further improved to 58.13% and 67.69%, if $\mathcal{H}_{fit}(S)$ and new elongation $\mathcal{E}(S)$ measure were applied separately. Combining both $\mathcal{E}(S)$ and $\mathcal{H}(S)$, the classification accuracy improves to 73.23%, while a better accuracy was obtained by combining $\mathcal{E}(S)$ and $\mathcal{H}_{fit}(S)$ instead: 75.91%. Notice that such an obtained classification rate (using only two simple features) is comparable to the accuracy of 77.44% obtained using the initial five descriptors. Furthermore, this result is also better than one obtained using only the first Hu moment invariant (63.36%, as indicated in [162]), as well as a rate of 70.99% obtained using the one version of $\mathcal{D}(S)$, as indicated in [162].

Chapter 6

Signature of shape utilizing its pixel coverage representation

A shape signature is usually defined as an arbitrary one-dimensional function representing a given two-dimensional shape. The signature, commonly, needs to capture most of the perceptual properties of the shape aimed to provide its unique representation. To provide its applicability in a wide spectrum of shape-based object analysis tasks, the signature should be invariant under rotation and translation transformations, but also robust to noise and other various distortions of a shape. Different shape signatures have been designed and developed in the literature over the years [40, 51, 101, 181, 182]. Some of them are based on *centroid distance, triangular centroid area, curvature scale space, farthest point distance signature, as well as complex network and spectral graph theory*. Among them, signature based on the *centroid distance* of the boundary points represents one of the commonly used and also popular signatures of a shape, whose popularity follows from its simple and intuitive definition, and its usability for further derivation of other important, yet popular object analysis tools such as, for example, Fourier descriptors of a shape (FDs) [182].

Commonly, the observed shapes are obtained in a process of crisp (i.e., binary) segmentation of the original image. As we have already mentioned, the loss of information about the original continuous shape, associated with the processes of discretization and binarization, can significantly affect the performance of the corresponding shape descriptor. Thus, for example, the reduced amount of shape information contained in the discrete binary representation consequently influences the reduction in precision and accuracy of the shape descriptor observed. The signature of a shape as a boundary-based descriptor is additionally sensitive to rotation and/or translation of shape, as well as to the presence of noise where a small amount of noise, especially on the boundary of a shape, can significantly disrupt the performances of

the signature considered.

This chapter presents a relatively recent approach to improving the performance of shape descriptors, computing them from a coverage based shape representation, instead of following the traditional crisp-based approach. The advantages of this approach over the binary shape representation have already been discussed in Chapter 4. The results obtained have confirmed that such a representation, rich in information of the original shape, can be of particular importance in the tasks where the precision and accuracy of the observed shape descriptors are of a crucial role. This motivates and additionally encourages further research of the development of a new approach for signature estimation based on the coverage shape representation. Thanks to these observations, we present, in this chapter, a new method for a shape signature estimate, originally introduced in [72], which iteratively improves the accuracy of the signature estimates. In each iteration, the algorithm produces a more accurate and more precise sub-pixel position of the boundary of a shape within the pixel, and consequently, a more accurate estimate of the signature observed. Statistical evaluation results verify the theoretically discussed performance of the approach considered, compared to both the standard crisp (i.e., binary) approach and one utilizing the average of computed signatures for all α-cuts of the observed fuzzy representation of shape [26]. The achieved performance improvements are related to increased precision and accuracy, improved invariance to translation and rotation, and increased robustness in the presence of noise as well.

6.1 Centroid distance signature

The signature of a shape based on the centroid distance represents one of the most commonly used signatures in diverse shape-based object analysis tasks. The *centroid distance signature* of a continuous shape is defined as a continuous function $r(t)$, assigning to each boundary point $(x(t), y(t))$, $t \in I \subset \mathbb{R}$, its (Euclidean) distance from the shape centroid $(\mathbf{x}, \mathbf{y})$, i.e.,

$$r(t) = \sqrt{(x(t) - \mathbf{x})^2 + (y(t) - \mathbf{y})^2}. \tag{6.1}$$

To ensure a unique reconstruction of the original object from the centroid distance signature, the object considered needs to be a star-shaped to its centroid [78]. As mentioned in Definition 2.11, the object S is *star-shaped* to the object point if, for each boundary point, the line segment that connects them is completely within the object S. A set of all the object points to which the observed object is star-shaped is called a *kernel* of the object (see Definition 2.12). If the centroid belongs to its kernel, then the signature based on the centroid distance can completely describe the observed object, enabling its unique reconstruction and retrieval.

Exploiting these observations, it can be noticed that for a star-shaped object whose centroid belongs to its kernel, it holds that any straight half-line starting from the shape centroid intersects the shape boundary at exactly one point. Such a property of star-shaped objects establishes a one-to-one correspondence between the interval $[0,2\pi)$ and the set of the boundary points. Moreover, such an established correspondence provides parameterization of the boundary by a parameter (i.e., an angle) from the interval $[0,2\pi)$, leading us to the following definition of the centroid distance signature of a continuous star-shaped object, which is originally introduced in [72].

Definition 6.1 *The centroid distance signature of a continuous star-shaped object S is a continuous function $r(\theta)$ which assigns the Euclidean length of the straight-line connecting the centroid of the shape $(\mathbf{x},\mathbf{y})$ and the boundary point $(x(\theta),y(\theta))$, to every angle $\theta \in [0,2\pi)$ as follows:*

$$r(\theta) = \sqrt{(x(\theta)-\mathbf{x})^2 + (y(\theta)-\mathbf{y})^2}, \tag{6.2}$$

where $\theta = \arctan\left(\frac{y(\theta)-\mathbf{y}}{x(\theta)-\mathbf{x}}\right)$, *denoting with* arctan *the four-quadrant inverse of the tangent, with an output values in the range of* $[0,2\pi)$.

However, the observed continuous object is usually represented with its digitization in the discrete space (e.g., $\mathbb{Z}^2$), where only its discrete representation is available. Following this, we are interested in a signature estimation of a continuous shape from its discrete (i.e., digital) representation, resulting usually from the process of digitizing the original continuous object. With this in mind, a continuous form signature from its discrete representation can be defined as an appropriate restriction of a continuous signature considering only a few appropriately selected θ samples from the interval $[0,2\pi)$.

Definition 6.2 *The signature of a discrete representation of a continuous star-shaped object S is a family of ordered pairs $(r(\theta_i),\theta_i)$, where $r(\theta_i)$ assigns the length of the straight-line, connecting the centroid of a shape $(\mathbf{x},\mathbf{y})$ and the boundary point $(x(\theta_i),y(\theta_i))$, to a discrete set of sample values $\theta_i \in [0,2\pi)$:*

$$r(\theta_i) = \sqrt{(x(\theta_i)-\mathbf{x})^2 + (y(\theta_i)-\mathbf{y})^2}, \qquad i = 1,2,\ldots,N, \tag{6.3}$$

where the sample values θ_i can be selected in several ways, and their choice, in general, could affect the shape description.

The sample values θ_i can be determined in advance (e.g., using equally distributed values of θ_i over the interval $[0,2\pi)$), or indirectly during the process of signature computation. In a situation where there is no information about the object position within the boundary pixel, it is natural to take the center of the pixel as a reasonable sample point. For the boundary pixels, we observe object pixels that contain at

least one background pixel in their corresponding neighborhood. Herein, we distinguish between the boundary pixels that are 4- or 8-connected with background pixels, depending on whether 4- or 8-neighborhood of the boundary pixel is observed, as defined respectively in (3.7) and (3.9). Thus, for the object with no holes, the boundary pixels that are 4-connected (i.e., 8-connected) with background pixel(s) are contained in 8-connected (i.e., 4-connected) object boundary. Based on this, we can observe the sample points within the pixels belonging to both 4- or 8-connected object boundaries. However, this choice of sample point necessarily introduces errors in signature computation, since it is unlikely that the continuous shape boundary passes through the center of the pixels. This further affects the performances of the observed signature in terms of achieved precision and accuracy, as well as its variance under translation and rotation. Also, as a boundary-based descriptor, the signature is particularly sensitive to noise, and the choice of sample points can further increase this weakness.

Following the approach proposed in [26], one way to reduce these negative effects is to compute the signature utilizing fuzzy discrete shape representation. In that work, the authors analyzed two methods for signature computation based on a fuzzy representation of a given shape. One of them, computed as the average of the signatures estimated for each α-cut, outperforms the other one which is also analyzed, as well as the method based on crisp shape representation. The improvements achieved are related only to increased precision, not accuracy, given that the authors did not address the bias of the signature estimate. This result comes from the fact that the proposed signatures utilize the center of the boundary pixel of each α-cut as a corresponding sample point of the shape boundary. Taking into account these considerations, in this chapter we present our new method, originally proposed in [72], for estimating the signature of a shape utilizing the coverage representation of the object we are considering. Contrary to the approach introduced in [26], our novel method is based on an estimation of the boundary position within the pixel, with sub-pixel precision. Such a sub-pixel position within the boundary pixel can be then utilized to improve signature in terms of increased accuracy and precision, as well as reduced variance concerning translation and rotation transformations. Besides, the proposed sub-pixel precise signature estimate is also more robust to the presence of noise, as compared to other already mentioned signature estimates.

6.2 Proposed signature with sub-pixel precision

Now, we present a novel method for estimating shape signature based on its pixel coverage representation. Under the assumption that the boundary of a shape within a pixel is locally straight, we can estimate the edge position with sub-pixel precision as a function of a pixel coverage a, for a given normal direction of edge $\mathbf{n} = (\cos\varphi, \sin\varphi)$. The discrete signature of a shape can then be computed by selecting a sample point on the estimated edge within the boundary pixel. To provide

a well defined inner point of the pixel, the midpoint of the estimated edge within the pixel can be one of the reasonable choices. Following this, according to definition of discrete shape signature (Definition 6.2), the *Pixel Coverage Shape Signature (PCSS)* is defined in [72] as follows.

Definition 6.3 *The pixel coverage signature of a star-shaped continuous object is a collection of pairs* $(r(\theta_i),\theta_i)$*, where* $r(\theta_i)$ *assigns the length of the straight-line connecting the shape centroid* $(\mathbf{x},\mathbf{y})$ *and the point* $(x_{mid}(\theta_i),y_{mid}(\theta_i))$*, to every angle* $\theta_i \in [0,2\pi)$*, where* $\theta_i = \arctan\left(\frac{y_{mid}(\theta_i)-\mathbf{y}}{x_{mid}(\theta_i)-\mathbf{x}}\right)$*,* $(x_{mid}(\theta_i),y_{mid}(\theta_i))$ *denotes a midpoint of the estimated edge (within the i-th boundary pixel along the boundary of the object), and*

$$r(\theta_i) = \sqrt{(x_{mid}(\theta_i)-\mathbf{x})^2+(y_{mid}(\theta_i)-\mathbf{y})^2},\ i = 1,2,\ldots,N. \tag{6.4}$$

Under the assumption that the boundary of the object, within the pixel centered at (x,y), is locally straight, the coordinates of the midpoint of the edge segment can be expressed as a function of a coverage value a and normal direction $\mathbf{n} = (\cos\varphi,\sin\varphi)$ as follows:

$$(x_{mid},y_{mid}) = (x+e_x(a,\varphi),y+e_y(a,\varphi)), \tag{6.5}$$

where, for all angles $\varphi \in [0,\pi/4]$, it holds that

$$(e_x(a,\varphi),e_y(a,\varphi)) \ = \ = \begin{cases} (\sqrt{\frac{a\tan\varphi}{2}}-\frac{1}{2},\sqrt{\frac{a}{2\tan\varphi}}-\frac{1}{2}) & ,0 \le a \le \frac{1}{2}\tan\varphi, \\ (a-\frac{1}{2},0) & ,\frac{1}{2}\tan\varphi \le a \le 1-\frac{1}{2}\tan\varphi, \\ (\frac{1}{2}-\sqrt{\frac{(1-a)\tan\varphi}{2}},\frac{1}{2}-\sqrt{\frac{1-a}{2\tan\varphi}}) & ,\ 1-\frac{1}{2}\tan\varphi \le a \le 1. \end{cases} \tag{6.6}$$

Besides, it can be noticed that the expressions in (6.6) are antisymmetric around $a = 0.5$, i.e., around the center of the pixel. Therefore, it is sufficient to consider only the coverage values ranging through the interval $[0,0.5]$, and also, due to symmetry in φ, it is enough to take only the angles $\varphi \in [0,\pi/4]$. Other cases can be obtained by changing the sign and/or swapping the coordinates of the midpoint if it is necessary.

6.3 Algorithm for signature estimation

In order to provide computation of the proposed pixel coverage shape signature (in this chapter it is referred to as PCSS) based on the equations (6.4) and (6.5), it is necessary to estimate the normal direction of the object boundary $\mathbf{n} = (\cos\varphi,\sin\varphi)$,

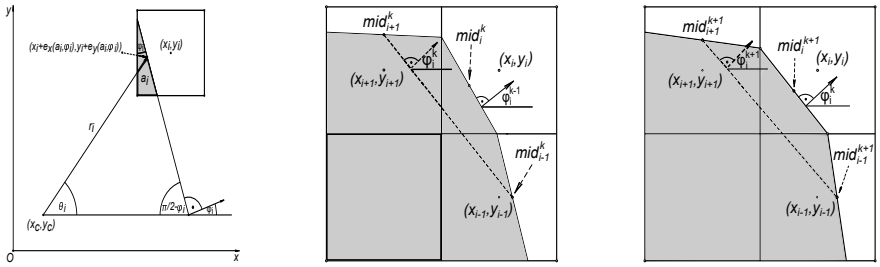


Figure 6.1: (a) Shape signature based on pixel coverage, PCSS, computed to the midpoint of the edge segment within a pixel (x_i, y_i), for a given coverage value a_i and normal direction φ_i. (b) and (c) Illustration of iterative estimation of the boundary position (i.e., midpoint $mid_i = (x_i + e_x(a_i, \varphi_i), y_i + e_y(a_i, \varphi_i))$ and normal direction φ_i in the two consecutive k-th and $(k+1)$-th iterations, utilizing the midpoints of the estimated edges within the two neighboring boundary pixels from the previous iteration.

within each boundary pixel (as illustrated in Fig. 6.1). For the boundary pixels, we consider those with assigned coverage values $a > 0$, which are 8-connected with pure background pixels, having the coverage value a equals to 0. In the case of the object without holes (e.g., star-shaped object), the pixel coverage based digitization produces 4-connected discrete boundary of one-pixel thickness which can be easily identified and parametrized as well.

In the following, we illustrate the main contribution of the work [72], i.e., we present an iterative algorithm for estimating the signature of a shape utilizing its coverage representation. Such an introduced algorithm is based on the current signature estimate which is then utilized to derive the normal direction and to improve the signature estimation itself as well. The algorithm proposed utilizes only the coverage values of the observed pixel and its two immediate neighbors (along the observed discrete boundary), and does not require the gradient estimate of the normal direction within each boundary pixel, as it has been considered in Chapter 4 for 3D Euclidean distance transforms.

The initial signature is given as a sequence of the distances computed from the shape centroid to the initial midpoints utilizing the normal direction initialized with $\varphi_i = \theta_i$. This further implies that the observed object edge, in each boundary pixel, is orthogonal to the vector connecting the centroid of shape and center of the boundary pixel. Edge position within each boundary pixel is then updated by estimating the straight line perpendicular to the straight line connecting the edge midpoints in the two neighboring boundary pixels. The current midpoints of the estimated object

edge within each boundary pixel are estimated using the signature estimation in the previous iteration. The new (i.e., updated) edge position (and, consequently, its new midpoint) is then estimated using the new normal direction estimate and the coverage value. Two consecutive iterations of such a defined signature computation process are illustrated in Fig. 6.1(b) and 6.1(c). Such a procedure is then repeated until the difference (i.e., distance) between two consecutive signature estimates (i.e., signature vectors) becomes less than a given tolerance, or until some predefined number of iterations is reached. Herein we rely on a maximum metric d_∞, defined (according to (3.5)) as follows:

$$d_\infty(\mathbf{r}^k, \mathbf{r}^{k-1}) = \max_{1 \le i \le N} |r_i^k - r_i^{k-1}|,\ k > 1, \tag{6.7}$$

where $\mathbf{r}^{k-1}$ and $\mathbf{r}^k$ are the signature estimates (i.e., vectors of estimated signatures) achieved in the $(k-1)$-th and k-th iteration, respectively.

Taking this into account, we are now able to present the main contribution of the work [72], i.e., a complete iterative algorithm for estimating signature of a continuous shape based on its pixel coverage based discrete (i.e., digital) representation. The algorithm presented is easy to parallelize if higher speed is necessary, given that each line assumes parallel or vectorized computation for all boundary pixels. Cyclic indexing is assumed when index $i \pm 1$ reaches end values of the indices assigned to the boundary pixels.

Algorithm 2

Input: *Pixel centers (x_i, y_i) observed in the counterclockwise along the object boundary, coverage based values a_i, $i = 1, \ldots, N$, tolerance $\varepsilon > 0$, and maximum number of iterations maxiter> 0.*
Output: *Pixel coverage signature PCSS, consisting of N ordered pairs (r_i, θ_i).*

$\varphi_i \leftarrow \arctan\left(\frac{y_i - \mathbf{y}}{x_i - \mathbf{x}}\right)$
for $k = 1, 2, \ldots,$ *maxiter*
  compute $(e_x(a_i, \varphi_i), e_y(a_i, \varphi_i))$ *according to* (6.6)
  $r_i^k \leftarrow \sqrt{(x_i + e_x(a_i, \varphi_i) - \mathbf{x})^2 + (y_i + e_y(a_i, \varphi_i) - \mathbf{y})^2}$
  $\theta_i^k \leftarrow \arctan\left(\frac{y_i + e_y(a_i, \varphi_i) - \mathbf{y}}{x_i + e_x(a_i, \varphi_i) - \mathbf{x}}\right)$
  if $k > 1 \wedge d_\infty(\mathbf{r}^k, \mathbf{r}^{k-1}) < \varepsilon$ *then*
    exit the for loop
  $\tilde{x}_i \leftarrow r_i^k \cos(\theta_i^k),\ \tilde{y}_i \leftarrow r_i^k \sin(\theta_i^k)$
  $\varphi_i \leftarrow \arctan\left(\frac{\tilde{x}_{i-1} - \tilde{x}_{i+1}}{\tilde{y}_{i+1} - \tilde{y}_{i-1}}\right)$
end for
$PCSS = \left\{(r_i^k, \theta_i^k)\right\},$ *for* $i = 1, 2, \ldots, N$

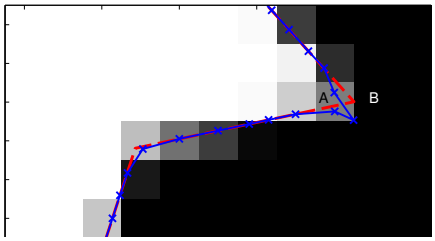


Figure 6.2: Example of adjusting the coverage of pixel intersected by more than one object edge. The pixel marked with A, having coverage of 0.52, is intersected twice by the exact object boundary (red dashed line), passing through the pixel B in between. Both edge midpoints of A are not placed correctly since they are close to the pixel center. If we assume that each edge cuts half of the background, and utilize a coverage value (for each edge separately) of 0.76, a better result (blue line) is achieved.

However, in the case of non-smooth objects, it can be possible to exist some boundary pixels that are intersected several times by continuous, non-smooth shape boundary. These pixels are visited more than once, and as a result, the algorithm proposed could produce sub-optimal estimates of the object edge within such pixels, which can further affect the performance of the signature estimated. Such a result follows from the fact that the exact position of the object boundary within the pixel cannot be correctly estimated, given that the assumption about local linearity of the object boundary is not satisfied, as required by Algorithm 2. Consequently, the appropriate adjustment of using the corresponding coverage value is required to improve the estimation of the object boundary within the pixel. One such a possibility may be splitting the coverage values assigned to pixels at sharp corners, as a part of the pre-processing step, and then utilizing new (i.e., updated) coverage values for each of the edge estimates separately, as illustrated in Fig. 6.2.

Adjustment

for all pixels j which appear $n_j > 1$ times along the boundary
  if $a_j < \frac{1}{2}$, let $a_j \leftarrow \frac{a_j}{n_j}$
  else, let $a_j \leftarrow 1 - \frac{1-a_j}{n_j}$

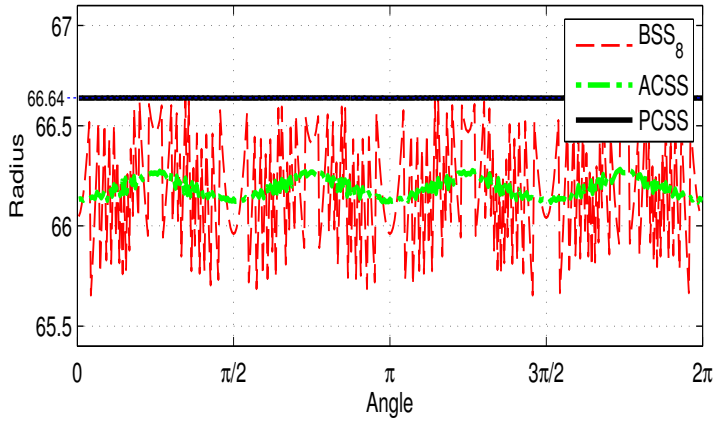


Figure 6.3: Shape signatures computed by BSS (8-connected boundary observed), ACSS and PCSS for a disc with a radius of 66.64 pixels. The errors of PCSS method are too small that they are hardly visible at this scale.

6.4 Experimental evaluation

In the following, we provide an evaluation of the performance of the proposed coverage based method for signature estimation, in comparison with methods based on a crisp (i.e., binary) shape representation (according to Definition 6.2), and utilizing α-cuts of the corresponding fuzzy shape representation, as proposed in [26]. These approaches herein will be referred to as BSS (Binary Shape Signature), ACSS (α-cut Shape Signature) and PCSS (Pixel Coverage Shape Signature), while the performance evaluation relates to improved precision and accuracy, robustness to noise, and invariance to translation and rotation.

In that context, we start with the initial test set consisting of discs, 6-cornered stars, and rectangles. To get a crisp shape representation of the observed objects, the Gauss digitization (according to Definition 2.16) of continuous shapes is performed, whereas the coverage shape representation is analytically derived, and as such, accurate to the level of floating-point operations. The observed fuzzy representation of shape is also based on the coverage of the pixels, indicating that ACSS and PCSS are applied to the same representation of a shape. For the exit criterion in Algorithm 2 we have used the tolerance level $\varepsilon = 0.0001$, or the maximal number of iterations maxiter=100.

Before we provide a more comprehensive illustration of the performance of PCSS, let first consider one simple but rather an illustrative example, which has motivated our proposed coverage based approach. The centroid distance signatures, estimated

by BSS, ACSS, and PCSS, of a disc with a radius of 66.64 pixels, are presented in Fig. 6.3. Relative to the exact centroid distance signature of a disc (i.e., a constant function $y = 66.64$), it can be noticed a very high accuracy and precision of the proposed PCSS method, in comparison with both more precise (with reduced angular variations), but non-accurate (i.e., biased) ACSS, and rather imprecise (with increased angular fluctuations) and also non-accurate BSS.

For further empirical evaluation, we observe the errors of the computed signatures of 100 randomly positioned discs and additionally rotated 6-cornered stars, of the same radius of 66.64 pixels. Histograms of errors (relative to exact continuous signature) for the objects observed are given in Fig. 6.4. For BSS we consider both 4- and 8-connected boundaries, while for ACSS only 8-connected boundary (as proposed in [26]) has been observed. Regarding our proposed PCSS method, we observe only the 4-connected boundary, given that it provides more sample points than the corresponding 8-connected counterpart. It can be noticed that significant improvement in the accuracy of the proposed PCSS has been achieved, together with a considerably smaller range of errors, compared to both BSS (with 4- and 8-connected boundary) and ACSS methods. The biases (i.e., lacks accuracy) for both BSS and ACSS are also visible, together with their variances (i.e., lacks precision).

For statistical evaluation of the signature estimates considered, we observe two quantitative measures to evaluate performances:

1) *root mean square error* (RMSE):

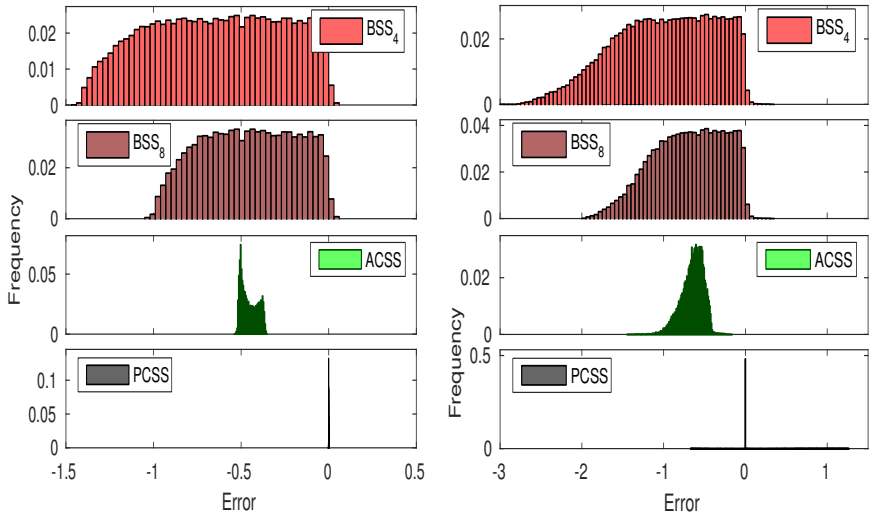
$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{x}_i - x_i)^2}\,, \tag{6.8}$$

2) *maximal absolute error* (MaxErr):

$$\mathrm{MaxErr} = \max_{1\le i\le N}|\hat{x}_i - x_i|\,, \tag{6.9}$$

where N represents the number of boundary pixels, while x_i and $\hat{x}_i$ represent the true and estimated values of the shape signature, respectively. Just to mention that measure (6.8) has been already considered in Chapter 4 to evaluate the performances of the proposed 3D EDTs.

For each radius ranging from 0 to 100 pixels, we observe 100 discs (as well as 100 stars) randomly translated within one pixel, whereas the stars are further randomly rotated. In the tasks of evaluation, as the corresponding ground truth, we observe the exact, i.e., analytically computed, signatures of the continuous objects we are considering. To provide an estimate of biased BSS and ACSS in terms of the achieved precision, we ignore their biases using the mean of the estimated signatures of 100 randomly translated discs of radius of 1000 pixels and subtract such a computed mean

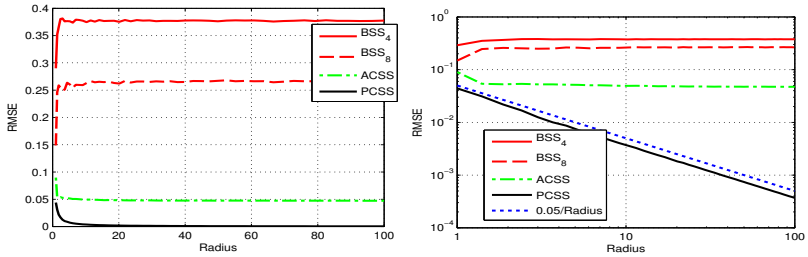


(a) Disk (b) Star

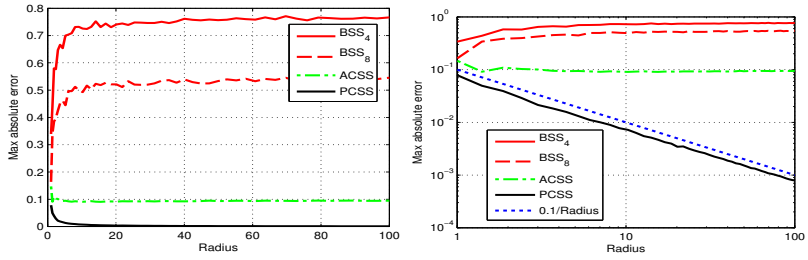
Figure 6.4: Histograms of errors for BSS, ACSS, and PCSS applied to a disc (a) and a star (b), of the radius of 66.64 pixels. Errors larger than 0.01 pixels are very rare for PCSS.

from the estimated signatures observed in the test, instead of the exact signatures. Plots of the computed RMSE and MaxErr values are presented in Fig. 6.5. It can be noticed that the proposed PCSS method, in general, outperforms both BSS and ACSS methods, particularly in the case of discs of increasing radii, whereas in the case of objects with non-smooth boundaries (e.g., stars) the limited improvement is still present.

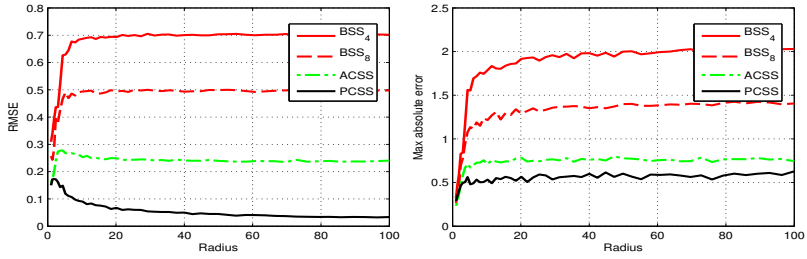
In the following experiment, we illustrate the rotational invariance of the observed signature estimates. In that sense, we observe three different rotations of a rectangle of 9×11 pixels, given in Fig. 6.6(b) by their coverage based representations. Plots of estimated signatures of these shapes using these three competitive methods are presented in Fig. 6.6(a). Each of the plots presents the estimated signatures of these three shapes by one of the three observed estimation methods given with differently labeled points using triangles, squares, and stars. To provide a better understanding of the obtained results, as well as the behaviour of the estimation methods observed, the exact signature of the rectangle is also presented in each plot as a blue continuous line. Notice also that all the computed signatures previously cyclically set to the same starting point (i.e., angle), indicating that ideally the signatures computed will coincide. It can be noticed that variance to translation and rotation is pretty visible



(a) RMSE disk (b) RMSE log-log



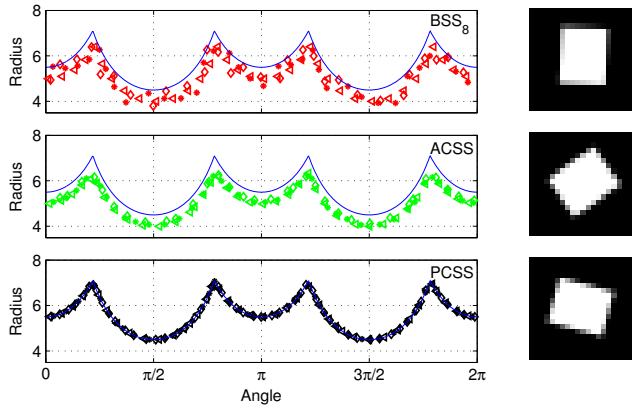
(c) MaxErr disk (d) MaxErr log-log



(e) RMSE star (f) MaxErr star

Figure 6.5: RMSE and MaxErr, measured in pixels, of BSS (4- and 8-connected), ACSS (8-connected) and PCSS, applied to discs (first and second row) and stars (third row) of increasing radii.

for BSS (as expected), as well as for ACSS, due to the effects of discretization. Deviations of the labeled points from the continuous blue line (i.e., lack of accuracy) and between the measurements (i.e., lack of precision) are clearly visible for both BSS and ACSS, whereas for PCSS these points are consistently (i.e., precisely) and correctly (i.e. accurately) distributed, indicating both high precision and high accuracy.

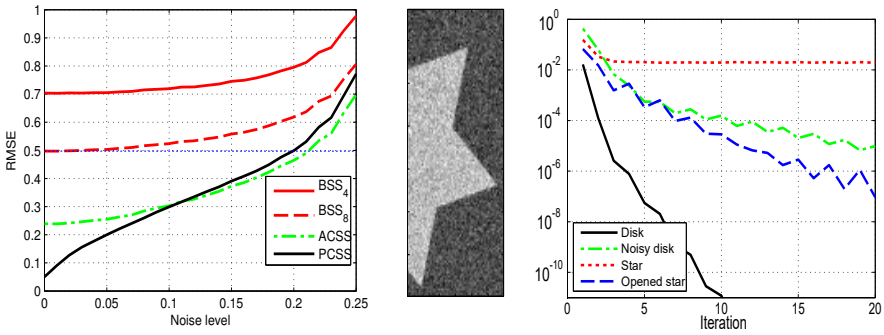


(a) (b)

Figure 6.6: Signatures estimated by BSS, ACSS, and PCSS for three rotations of rectangle. Points of the different signatures are labeled by triangles, squares, and stars. Coverage representations of the rectangles are shown to the right.

Besides, we have also evaluated robustness to the noise presence of the proposed PCSS relative to both BSS and ACSS. For this purpose, we observe the signatures computed for a star when increasing grades of Gaussian noise, centered in zero and standard deviation ranging through $[0, 0.25]$, is added. Each object generated is then segmented by Otsu thresholding method [107] to provide crisp shape representation (for BSS), or using the fast soft thresholding technique to provide coverage shape representation (for ACSS and PCSS), as proposed in [147]. The computed RMSE values of the estimated signatures for each of the observed objects are in Fig. 6.7. It can be noticed that both estimation methods ACSS and PCSS behave better in comparison with the crisp (i.e., noise-free) BSS, given as a blue dotted line, even though coverage representation is degraded up to 0.2 level of noise. Regarding the observed BSS methods utilizing 4-connected, as well as 8-connected object boundary, the presented plots indicate their insufficient robustness in such tasks.

For an illustration of the algorithm convergence, the plots representing the differences (i.e., distances) between two successive signature estimates $d_\infty(\mathbf{r}^k, \mathbf{r}^{k-1})$ are shown in Fig. 6.7(c). It can be noticed that the results obtained non-monotonically decrease for shapes having no sharp corners (e.g., discs). Such results are also consistent with our expectation, given that the main assumption about the local linearity of the boundary can be considered to be pretty reasonable for such shapes. Contrary to this, in the case where such an assumption cannot be considered to be reasonable (e.g., in the case of stars), the algorithm proposed terminates when the predefined



(a) (b) (c)

Figure 6.7: RMSE for the methods considered for increasing levels of Gaussian noise. Both PCSS and ACSS perform better even if up to 20% noise is added than what a noise-free crisp approach (blue dotted line) does. (b) Part of the observed object with 20% noise added. (c) Asymptotic behaviour of the sequence $d_\infty(\mathbf{r}^k, \mathbf{r}^{k-1})$, $k \in \mathbb{N}$, for different cases.

number of iterations is reached. Nevertheless, if the sharp angles of the star could be smoothed by applying the appropriate morphological operations as a corresponding pre-processing step, for example, the proposed algorithm converges as well. This result is illustrated observing blue dashed line plotted in Fig. 6.7(c).

Chapter 7

Fuzzy squareness: a new approach to describing the shape

Herein we present a new fuzzy squareness measure to quantify to what extent a given fuzzy shape matches a fuzzy square. This fuzzy shape-based measure is naturally defined and theoretically well-founded, resulting in that its behaviour can be understood and predicted in advance. As already said, there is an increasing need for studying different descriptors of shapes, as well as defining new methods for measuring both existing and new characteristics of shapes. This latter comes from the fact that no measure satisfies all expected properties in all applications since a shape measure that performs well in one task does not have to be equally efficient in another task. Based on this, several shape descriptors has been defined so far, among which there are those for which several different measures have been presented until now. In that context, in Chapter 5 it has been introduced a new shape-based descriptor and also presented a new method for measuring how much hexagonal is a given shape considered. Besides, we have presented that such a shape-based measure can provide several useful consequences, i.e., new shape measures, which have also been proved useful in various tasks and applications for analyzing objects. In this chapter, we follow the same line as in Chapter 5 for defining a new shape-based descriptor of an object, and introduce a new method for its numerical characterization.

In that sense, let us recall once again that the shape represents one of the basic components of the object, together with color and structure, that has many characteristics that can be numerically evaluated and, consequently, used to characterize objects. Also, the shape is usually represented as a bounded planar region, which

0.8178	0.7798	0.6817	0.6756	0.6567	0.6231	0.0011	0.0066	0.0048	0.0170
(0.7154)	(0.6536)	(0.4600)	(0.4530)	(0.4475)	(0.4311)	(0)	(0)	(0)	(0)

Figure 7.1: Several synthetic shapes, and their measured $\mathcal{FS}_q(S)$ and $\mathcal{FS}_{fit}(S)$. $\mathcal{FS}_{fit}(S)$ values are in the brackets.

corresponds to the spatial extent of the object when its visual characteristics, such as color and texture, have been removed from consideration. In this chapter, we assume that each appearing shape will be considered to be spatial fuzzy subset having a membership function that illustrates to what extent the spatial element belongs to an object. As it has been already mentioned in Chapter 2, the advantages of presenting objects as the appropriate fuzzy sets are of particular importance when the uncertainty and vagueness, present in an image, represent the unavoided result of many imaging techniques, or when image data we are considering is of low quality. Thanks to these observations, we find that the concept of fuzzy sets represents a natural approach for preserving the important information about an imaged object relating to a fuzziness present in an image.

Being a specific feature of the object, a fuzzy squareness has an obvious geometric interpretation (i.e., meaning), which enables us to introduce it by applying some commonly known approaches to defining object descriptors. One such approach, which has already been mentioned in Chapter 5, consists of fitting the model to the shape considered, and evaluating the extent of fit as the corresponding measure of shape [121]. In the case of fuzzy shapes, this approach can be expressed as follows: consider a fuzzy square $\mathcal{FS}(S)$ that in the best way fits the observed fuzzy shape S, and then define a fuzzy squareness as the degree of similarity between $\mathcal{FS}(S)$ and S. Among the several possible different choices for the corresponding fitted fuzzy square, in this chapter we have decided to observe a fuzzy square whose support and core are of equal areas as the support and core of S, and whose centroids are coincident with the centroids of the support and core of S. Exploiting this approach, we come to fuzzy squareness measure $\mathcal{FS}_{fit}(S)$ defined as

$$\mathcal{FS}_{fit}(S) = \frac{Area\,(S \cap \mathcal{FS}(S))}{Area\,(S \cup \mathcal{FS}(S))}, \tag{7.1}$$

denoting with $\mathcal{FS}(S)$ rotation of a fuzzy square around the centroid so that the area of $S \cap \mathcal{FS}(S)$ is as large as possible. Taking into account such a definition of fuzzy

squareness, it is straightforward to verify several of its basic and also very important properties:

P1) The measured fuzzy squareness $\mathcal{FS}_{fit}(S)$ is a number from $[0,1]$;

P2) The fuzzy squareness $\mathcal{FS}_{fit}(S)$ returns 1 if and only if for S is a fuzzy square;

P3) $\mathcal{FS}_{fit}(S)$ is invariant under rotation, translation, and scaling transformations for fuzzy objects.

Notice that among the properties listed, the property P1 deserves special consideration. In fact, considering the property P1, it follows that there exist shapes whose measured fuzzy squareness equals 0. As an illustration of this property of $\mathcal{FS}_{fit}(S)$ we can observe the last four shapes in Fig. 7.1. It can be noticed that such a property of $\mathcal{FS}_{fit}(S)$ can be considered to be its essential weakness since we are interested in defining a new shape based measure which evaluates each non-zero area shape by degree of its fuzzy squareness corresponding to value ranging through the interval $(0,1]$. This further implies that a well-defined fuzzy squareness measure should satisfy all three listed properties, requiring slightly modification relating to property P1, to provide that the measured squareness for a non-zero fuzzy shape is greater than 0. In this chapter, we will present a new fuzzy shape squareness measure taking all the values from $(0,1]$, with the largest fuzzy squareness equals 1 if and only if the shape considered is a fuzzy square. Such a defined squareness has a few elementary properties that have been discussed from both the theoretical and experimental point of view. The new fuzzy squareness measure is also invariant to similarity transformations. Several various experiments to experimentally validate the behaviour of the new measure, and to verify all the theoretically proven results are also shown. Effectiveness and usefulness of the new fuzzy squareness measures have been demonstrated in diverse object analysis tasks performed on several widely studied image datasets.

The chapter presented is organized as follows. The next section provides the basic denotations and assumptions needed for deriving the main result of the chapter. A new fuzzy squareness measure is introduced in Section 7.2. Few desirable properties of the new squareness measure are also given in this section. Synthetic and real image experiments to illustrate the behaviour of the new fuzzy squareness measure, as well as the theoretically proven results from the previous section are in Section 7.3. Benefits and importance of the new shape measure in diverse classification tasks are illustrated in Section 7.4.

7.1 Preliminaries

This section introduces several basic definitions and notations used in this chapter. Besides, we will also introduce several important assumptions that will not restrict

the importance of the main result of this chapter but will enable us that its theoretical foundation are well understood, and its derivation to be mathematically correct. Following this, we briefly recall some of the underlying definitions already introduced in Chapter 2 that will be of particular importance in our derivation. Also, unless otherwise stated, we will assume that all the appearing objects are represented as bounded planar regions defined on Euclidean plane $\mathbb{R}^2$.

Definition 7.1 *A fuzzy subset S defined on the reference set* $\mathbb{R}^2$ *is a set of the ordered pairs* $S = \{((x,y), \mu_S(x,y) \mid (x,y) \in \mathbb{R}^2\}$ *with* $\mu_S : X \to [0,1]$ *representing a membership function of S, and* $\mu_S(x,y)$ *denoting a membership level of* $(x,y) \in \mathbb{R}^2$ *to a fuzzy set* $S \subset \mathbb{R}^2$.

Definition 7.2 *The* α*-cut of a fuzzy subset* $S \subset \mathbb{R}^2$*, for all* $\alpha \in (0,1]$*, is a crisp subset* $S^\alpha = \{(x,y) \in \mathbb{R}^2 \mid \mu_S(x,y) \geq \alpha\}$.

Definition 7.3 *The support of a fuzzy subset* $S \subset \mathbb{R}^2$ *is the crisp subset* $Supp(S) = \{(x,y) \in \mathbb{R}^2 \mid \mu_S(x,y) > 0\}$.

Definition 7.4 *The core of* $S \subset \mathbb{R}^2$ *is defined as the crisp subset* $Core(S) = \{x \in \mathbb{R}^2 \mid \mu_S(x) = 1\}$.

Besides, by the term *fuzzy point* we mean a special fuzzy subset on $\mathbb{R}^2$ defined in [24] in the following way:

Definition 7.5 *A fuzzy point at* $(a,b) \in \mathbb{R}^2$ *is a fuzzy subset* $\bar{P}(a,b)$ *defined by its membership function as follows:*

i) $\mu_{\bar{P}(a,b)}$ *is upper semi-continuous,*

ii) $\mu_{\bar{P}(a,b)}(x,y) = 1$ *if and only if* $(x,y) = (a,b)$*,*

iii) $\bar{P}(a,b)^\alpha$ *is a compact and convex subset of* $\mathbb{R}^2$*, for all* $\alpha \in [0,1]$.

Given that we are interested in designing a new fuzzy shape-based measure having most of the desirable properties that each well-defined measure of shape should satisfy, invariance concerning geometrical transformations of fuzzy objects, e.g., translation, rotation, and scaling, is unquestionably one of them. In this regard, we provide herein their formal definition in the case of fuzzy objects (i.e., fuzzy sets) defined on the Euclidean plane $\mathbb{R}^2$ as the corresponding reference set.

Definition 7.6 *A fuzzy subset* $S_{(a,b)} \subset \mathbb{R}^2$ *is translation of a fuzzy subset* $S \subset \mathbb{R}^2$ *for a given vector* (a,b) *if there exists a translation of the Euclidean plane* $\tau_{(a,b)} : \mathbb{R}^2 \to \mathbb{R}^2$*, defined as* $\tau_{(a,b)}(x,y) = (x+a, y+b)$*, if for all* $(x,y) \in \mathbb{R}^2$ *it holds:*

1) $Supp(S_{(a,b)}) = \tau_{(a,b)}(Supp(S))$,

2) $\mu_{S_{(a,b)}}(x+a, y+b) = \mu_S(x,y)$.

Definition 7.7 *A fuzzy subset* $S_\alpha \subset \mathbb{R}^2$ *is a rotation of a fuzzy subset* $S \subset \mathbb{R}^2$ *around the origin* $O(0,0)$ *by an angle* $\alpha \in [0, 2\pi)$ *if there exists a rotation of the Euclidean plane* $\mathcal{R}_{O,\alpha} : \mathbb{R}^2 \to \mathbb{R}^2$ *around the origin* $O(0,0)$ *by an angle* $\alpha \in [0, 2\pi)$*, defined as* $\mathcal{R}_{O,\alpha}(x,y) = (x \cdot \cos\alpha - y \cdot \sin\alpha, x \cdot \sin\alpha + y \cdot \cos\alpha)$*, if for all* $(x,y) \in \mathbb{R}^2$ *it holds:*

1) $Supp(S_\alpha) = \mathcal{R}_{O,\alpha}(Supp(S))$,

2) $\mu_{S_\alpha}(x \cdot \cos\alpha - y \cdot \sin\alpha, x \cdot \sin\alpha + y \cdot \cos\alpha) = \mu_S(x,y)$.

Definition 7.8 *A fuzzy subset* $\lambda \cdot S \subset \mathbb{R}^2$ *is a scaling of a fuzzy subset* $S \subset \mathbb{R}^2$ *by a real number* $\lambda > 0$ *if there exists a homothety of the Euclidean plane* $\mathcal{H}_{O,\lambda} : \mathbb{R}^2 \to \mathbb{R}^2$ *with the center in the origin* $O(0,0)$ *and the coefficient* $\lambda > 0$*, defined as* $\mathcal{H}_{O,\lambda}(x,y) = (\lambda \cdot x, \lambda \cdot y)$*, if for all* $(x,y) \in \mathbb{R}^2$ *it holds:*

1) $Supp(\lambda \cdot S) = \mathcal{H}_{O,\lambda}(Supp(S))$,

2) $\mu_{\lambda \cdot S}(\lambda \cdot x, \lambda \cdot y) = \mu_S(x,y)$.

Of particular importance for our derivation will be a fuzzy shape defined as a planar region whose points have membership values dependent only on their distance to the centroid of a shape. Considering this, we will use a d_1-distance defined for two given planar points $A(x_1, y_1)$ and $B(x_2, y_2)$ as follows

$$d_1(A,B) = d_1((x_1,y_1),(x_2,y_2)) = |x_1 - x_2| + |y_1 - y_2|. \tag{7.2}$$

In this respect, let us notice that the set of all $(x,y) \in \mathbb{R}^2$ for which d_1-distance from a given point is smaller than a given constant represents a *square* (i.e., d_1-disc). Exploiting this and adopting the definition of a fuzzy disc, introduced in [17], we will come to a new definition of a fuzzy square whose membership function depends exclusively on d_1-distance to its centroid. For this purpose, we will follow the approach used in [146], suitably adapted to defining a fuzzy square. Before that, we introduce the following definition.

Definition 7.9 *Let* $\bar{\mu} : \mathbb{R} \to [0,1]$ *be a non-increasing function, and denote with* $d_1((x,y),(\mathbf{x},\mathbf{y}))$ *a* d_1*-distance of* (x,y) *to a given point* $(\mathbf{x},\mathbf{y}) \in \mathbb{R}^2$*. A fuzzy square* $\mathcal{FS}$ *having centroid* $(\mathbf{x},\mathbf{y})$ *is a fuzzy set whose membership function* $\mu_{\mathcal{FS}} : \mathbb{R}^2 \to [0,1]$ *is defined as*

$$\mu_{\mathcal{FS}}(x,y) = \bar{\mu}(d_1((x,y),(\mathbf{x},\mathbf{y}))). \tag{7.3}$$

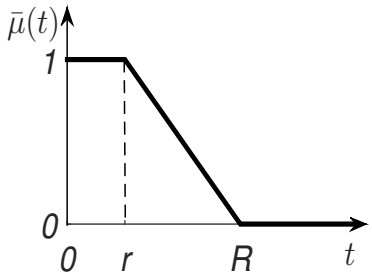


Figure 7.2: Plot of function $\bar{\mu}(t)$ defined in (7.4) for $r,R \in \mathbb{R}$ and $0 \leq r < R$.

In the following, unless otherwise stated, it is assumed that the centroid of all the appearing fuzzy sets coincides with the origin. The assumption made in this way is not a limitation in applications, given that the shape of a fuzzy object is invariant to translation. Herein, we will consider a fuzzy square with membership function determined uniquely by a piece-wise linear non-increasing function $\bar{\mu}$ defined as

$$\bar{\mu}(t) = \begin{cases} 1 & , \quad t \leq r, \\ \dfrac{t-R}{r-R} & , \quad r < t < R, \\ 0 & , \quad t \geq R, \end{cases} \tag{7.4}$$

where $0 \leq r < R$ and $r,R \in \mathbb{R}$ (see Fig. 7.2). It is easy to notice that the function $\bar{\mu}(t)$ together with Definition 7.9 generates a fuzzy square consisting of the points with membership values equal to 1 within d_1-distance r from the origin, equal to 0 out of d_1-distance R, and strictly between 0 and 1 within d_1-distance r and R from the origin. This brings us to the conclusion that the core and support of such a generated fuzzy square are the crisp squares (i.e., d_1-discs), centered at the origin, with a radius equals to r and R, respectively. Based on this, we can define a fuzzy square as a fuzzy set $\mathcal{FS}(r,R)$ with membership function $\mu_{\mathcal{FS}(r,R)} : \mathbb{R}^2 \to [0,1]$ defined as follows:

$$\mu_{\mathcal{FS}(r,R)}(x,y) = \begin{cases} 1 & , \quad |x|+|y| \leq r, \\ \dfrac{|x|+|y|-R}{r-R} & , \quad r < |x|+|y| < R, \\ 0 & , \quad |x|+|y| \geq R. \end{cases} \tag{7.5}$$

In a special case, for $r = 0$, we have a fuzzy point (according to Definition 7.5) with

membership function

$$\mu_{\mathcal{FS}(0,R)}(x,y) = \frac{|x|+|y|-R}{-R}\ , \ \text{ for } |x|+|y| \leq R;$$

while, for $r = R$, we get a crisp square with characteristic function $\mu_{\mathcal{FS}(r,R)}(x,y) = 1$, for $|x|+|y| \leq r$.

7.2 Measuring fuzzy squareness

Now, we can derive the main result of the chapter, i.e., a new fuzzy shape squareness measure, originally introduced in [70]. Also, several useful and desirable properties of such a shape measure are also illustrated and theoretically verified. In that context, let first be defined the auxiliary fuzzy square $\mathcal{FS}(S)$ for a given fuzzy shape S using the membership function $\mu_{\mathcal{FS}(S)}(x,y)$ given as

$$\mu_{\mathcal{FS}(S)}(x,y) = \begin{cases} 1 & , \quad |x|+|y| \leq r(S), \\ \dfrac{|x|+|y|-R(S)}{r(S)-R(S)} & , \quad r(S) < |x|+|y| < R(S), \\ 0 & , \quad |x|+|y| \geq R(S), \end{cases} \tag{7.6}$$

where the quantities $r(S)$ and $R(S)$ are defined as

$$r(S) = \frac{\sqrt{2}}{2}\left(Area(S^{\mathbb{C}})\right)^{1/2},$$

$$R(S) = \frac{\sqrt{2}}{2}\left(Area(S^{\mathbb{S}})\right)^{1/2}, \tag{7.7}$$

with $S^{\mathbb{C}}$ and $S^{\mathbb{S}}$ denoting the core and support of S, respectively. Based on this, for such auxiliary fuzzy square $\mathcal{FS}(S)$, the following applies:

- $\mathcal{FS}(S)$ is dependent only on a fuzzy shape S, and does not change under the rotation of S for an arbitrary angle. This follows from the fact that $r(S)$ and $R(S)$ are expressed by areas of the support and core of S, which are such invariants.
- The quantities $r(S)$ and $R(S)$ are defined in the way that the support and core of $\mathcal{FS}(S)$ and S are of equal area, i.e., $Area(S^{\mathbb{S}}) = Area(\mathcal{FS}(S)^{\mathbb{S}})$ and $Area(S^{\mathbb{C}}) = Area(\mathcal{FS}(S)^{\mathbb{C}})$.

Further, of particular interest for our derivation will be the quantity

$$\max_{\theta\in[0,2\pi)} \iint\limits_{S^{\mathbb{S}}_{\theta}} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy \tag{7.8}$$

where S_θ represents rotation of fuzzy shape S for an angle θ around its centroid. In the following, we will present that such a quantity achieves the largest possible value if and only if S is a fuzzy square. This result will further provide us with a new fuzzy squareness measure of a fuzzy shape. Before that, we prove the following theorem where for proving some statements we follow the approach similar to the one already used in Chapter 5.

Theorem 7.1 *Let be given a fuzzy shape S having the centroid coincident with the origin. Besides, let denote with S_θ a shape S rotated for an arbitrary angle θ around its centroid. Then the following applies:*

$$\frac{\iint\limits_{S^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\,dxdy}{Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}}) \cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})} \leq \frac{1}{3}; \tag{7.9}$$

$$\frac{\iint\limits_{S^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\,dxdy}{Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}}) \cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})} = \frac{1}{3}$$

$$\Leftrightarrow S = \mathcal{FS}(S); \tag{7.10}$$

$$\frac{\max\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\,dxdy}{Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}}) \cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})} = \frac{1}{3}$$

$$\Leftrightarrow S \text{ is a fuzzy square.} \tag{7.11}$$

Proof 7.1 *Let S be a fuzzy shape according to conditions stated in the theorem. Also, let $\mathcal{FS}(S)$ be the auxiliary fuzzy square, centered at the origin, with core and support having the vertices $(-r(S),0)$, $(0,-r(S))$, $(r(S),0)$, $(0,r(S))$, and $(-R(S),0)$, $(0,-R(S))$, $(R(S),0)$, $(0,R(S))$, respectively. Based on this, it is easy to validate that the support and core of S and $\mathcal{FS}(S)$ have the same area, as well as the set differences $S^{\mathbb{S}} \setminus \mathcal{FS}(S)^{\mathbb{S}}$ and $\mathcal{FS}(S)^{\mathbb{S}} \setminus S^{\mathbb{S}}$, and $S^{\mathbb{C}} \setminus \mathcal{FS}(S)^{\mathbb{C}}$ and $\mathcal{FS}(S)^{\mathbb{C}} \setminus S^{\mathbb{C}}$. Given that, according to* (7.6)*, $\mu_{\mathcal{FS}(S)}(x,y) = 0$ for all $(x,y) \in S^{\mathbb{S}} \setminus \mathcal{FS}(S)^{\mathbb{S}}$, we prove* (7.9)

as follows:

$$
\begin{aligned}
&\iint_{S^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy \;=\\
&= \iint_{S^{\mathbb{S}}\setminus\mathcal{FS}(S)^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy \;+\; \iint_{S^{\mathbb{S}}\cap\mathcal{FS}(S)^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy\\
&\leq \iint_{\mathcal{FS}(S)^{\mathbb{S}}\setminus S^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy \;+\; \iint_{\mathcal{FS}(S)^{\mathbb{S}}\cap S^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy\\
&= \iint_{\mathcal{FS}(S)^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy\\
&= \iint_{\mathcal{FS}(S)^{\mathbb{S}}\setminus\mathcal{FS}(S)^{\mathbb{C}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy + \iint_{\mathcal{FS}(S)^{\mathbb{C}}} \mu_{\mathcal{FS}(S)}(x,y)dxdy\\
&= \frac{1}{3}\left(Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}})\cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})\right).
\end{aligned}
\tag{7.12}
$$

The proof for (7.10) *implies from the fact that equality in* (7.12) *holds true if and only if* $Area(S^{\mathbb{S}} \setminus \mathcal{FS}(S)^{\mathbb{S}}) = Area(\mathcal{FS}(S)^{\mathbb{S}} \setminus S^{\mathbb{S}})) = 0$*, i.e., if and only if the fuzzy shapes* S *and* $\mathcal{FS}(S)$ *are the same.*

To prove (7.11)*, it should be noticed that* $\iint_{S^{\mathbb{S}}_{\theta}} \mu_{\mathcal{FS}(S)}(x,y)\; dxdy$ *varies, if the shape* S *is rotated around the origin for all orientations from* $[0,2\pi)$*. In that context, if we denote with* θ_0 *an angle maximizing the quantity* $\iint_{S^{\mathbb{S}}_{\theta}} \mu_{\mathcal{FS}(S)}(x,y)\; dxdy$ *over all rotations from* $[0,2\pi)$*, i.e.,*

$$
\iint_{S^{\mathbb{S}}_{\theta_0}} \mu_{\mathcal{FS}(S)}(x,y)\; dxdy \;=\; \max_{\theta\in[0,2\pi)} \iint_{S^{\mathbb{S}}_{\theta}} \mu_{\mathcal{FS}(S)}(x,y)\; dxdy. \tag{7.13}
$$

Then, assuming that the left side of (7.11) *is true, the following applies*

$$
\frac{\displaystyle\iint_{S^{\mathbb{S}}_{\theta_0}} \mu_{\mathcal{FS}(S)}(x,y)\; dxdy}{Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}})\cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})} \;=\; \frac{1}{3}. \tag{7.14}
$$

This establishes the proof of (7.11)*, given that* (7.14) *and* (7.10) *provide that fuzzy shapes S_{θ_0} and $\mathcal{FS}(S)$ are the same, implying that S must be a fuzzy square.* □

Now, based on the results of Theorem 7.1, we come to the fact that the following quantity

$$\frac{\max\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\, dxdy}{Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}}) \cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})}$$

is invariant concerning the rotation of S, and reaches the maximal value $1/3$ if and only if the shape S is a fuzzy square. Considering this, we have that the quantity

$$3 \cdot \frac{\max\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\, dxdy}{Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}}) \cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})} \tag{7.15}$$

can be used as a new shape measure. In this chapter, this measure will be named *fuzzy squareness shape measure*, and its formal definition is given as follows.

Definition 7.10 *Let a fuzzy shape S, centered in the origin, be given. A fuzzy squareness shape measure $\mathcal{FS}_q(S)$ of shape S is defined as*

$$\mathcal{FS}_q(S) = 3 \cdot \frac{\max\limits_{\theta\in[0,2\pi)} \iint\limits_{S_\theta^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\, dxdy}{Area(S^{\mathbb{S}}) + \sqrt{Area(S^{\mathbb{S}}) \cdot Area(S^{\mathbb{C}})} + Area(S^{\mathbb{C}})}, \tag{7.16}$$

where S_θ denotes rotated fuzzy shape S for an angle θ around its centroid.

Several basic and also desirable properties of the new fuzzy squareness shape measure $\mathcal{FS}_q(S)$ are summarized in the following theorem.

Theorem 7.2 *Given a fuzzy shape S, centered at the origin, the following statements are true:*

(a) $\mathcal{FS}_q(S) \in (0,1]$, for all shapes S*;*

(b) $\mathcal{FS}_q(S) = 1 \Leftrightarrow S$ is a fuzzy square*;*

(c) $\mathcal{FS}_q(S)$ is invariant to similarity transformations*;*

(d) if S is a crisp square, then $\mathcal{FS}_q(S) = 1$;

(e) if S is a fuzzy point whose all α-cuts are crisp squares, then $\mathcal{FS}_q(S) = 1$.

Proof 7.2 *Items (a) and (b) follow straightforwardly from* (7.9) *and* (7.11) *provided by Theorem 7.1. Item (c) follows from fact that* $Area(S^{\mathbb{S}})$ *and* $Area(S^{\mathbb{C}})$*, necessary for computation* $\mathcal{FS}_q(S)$*, as well as* $\max\limits_{\theta\in[0,2\pi)} \iint\limits_{S^{\mathbb{S}}_\theta} \mu_{\mathcal{FS}(S)}(x,y)dxdy$ *are rotational invariants. Following this, we have that* $\mathcal{FS}_q(S)$ *is also rotational invariant. Translation invariance comes from the fact that the origin and shape centroid are coincident. In order to prove invariance to scaling, let first assume that S is scaled by factor* λ *to shape* $\lambda \cdot S = \{(\lambda \cdot x, \lambda \cdot y) \mid (x,y) \in S\}$*. Exploiting this, we have the following:*

$$Area((\lambda \cdot S)^{\mathbb{S}}) \;=\; \lambda^2 \cdot Area(S^{\mathbb{S}}), \tag{7.17}$$

$$Area((\lambda \cdot S)^{\mathbb{C}}) \;=\; \lambda^2 \cdot Area(S^{\mathbb{C}}), \tag{7.18}$$

$$\max_{\theta\in[0,2\pi)} \iint\limits_{(\lambda\cdot S_\theta)^{\mathbb{S}}} \mu_{\mathcal{FS}(\lambda\cdot S)}(x,y)dxdy \;=$$

$$= \lambda^2 \cdot \max_{\theta\in[0,2\pi)} \iint\limits_{S^{\mathbb{S}}_\theta} \mu_{\mathcal{FS}(S)}(x,y)dxdy, \tag{7.19}$$

and, consequently,

$$\mathcal{FS}_q(\lambda \cdot S) \;=$$

$$= 3 \cdot \frac{\max\limits_{\theta\in[0,2\pi)} \iint\limits_{(\lambda\cdot S_\theta)^{\mathbb{S}}} \mu_{\mathcal{FS}(\lambda\cdot S)}(x,y)\; dxdy}{Area((\lambda \cdot S)^{\mathbb{S}}) + \sqrt{Area((\lambda \cdot S)^{\mathbb{S}}) \cdot Area((\lambda \cdot S)^{\mathbb{C}})} + Area((\lambda \cdot S)^{\mathbb{S}})}$$

$$= 3 \cdot \frac{\lambda^2 \cdot \max\limits_{\theta\in[0,2\pi)} \iint\limits_{S^{\mathbb{S}}_\theta} \mu_{\mathcal{FS}(S)}(x,y)\; dxdy}{\lambda^2 \cdot Area(S^{\mathbb{S}}) + \lambda^2 \cdot \sqrt{Area(S^{\mathbb{S}}) \cdot Area(S^{\mathbb{C}})} + \lambda^2 \cdot Area(S^{\mathbb{C}})}$$

$$= \mathcal{FS}_q(S),$$

giving that $\mathcal{FS}_q(S)$ *is invariant to scaling transformation. Statement (d) follows from* (7.16)*, and fact that core and support of S coincide, resulting that* $\mu_{\mathcal{FS}(S)}(x,y) = 1$*, for*

all $(x,y) \in S^{\mathbb{S}}$. *Then, we have:*

$$\mathcal{FS}_q(S) = 3 \cdot \frac{\max\limits_{\theta\in[0,2\pi)} \iint\limits_{S^{\mathbb{S}}_\theta} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy}{3 \cdot Area(S^{\mathbb{S}})} = \frac{\iint\limits_{S^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy}{Area(S^{\mathbb{S}})} = 1.$$

Finally, statement (e) follows from the fact that for a fuzzy square with a single-element core S it holds that $Area(S^{\mathbb{C}}) = 0$ *and* $\iint\limits_{S} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy = \frac{1}{3} Area(S^{\mathbb{S}})$, *i.e.,*

$$\mathcal{FS}_q(S) = 3 \cdot \frac{\max\limits_{\theta\in[0,2\pi)} \iint\limits_{S^{\mathbb{S}}_\theta} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy}{Area(S^{\mathbb{S}})} = 3 \cdot \frac{\iint\limits_{S^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy}{Area(S^{\mathbb{S}})} = 1.$$

This completes the proof of the whole theorem. □

Regarding the numerical algorithm for computing $\mathcal{FS}_q(S)$, it can be noticed that Definition 7.10 provides the algorithm consisting of rotating the fuzzy shape S around the origin, and evaluating the quantity $\iint\limits_{S^{\mathbb{S}}_\theta} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy$, for all θ through the interval $[0,2\pi)$. To make a computation of such quantities faster, we suggest using the appropriate geometric moments from (5.2), applied to suitably defined subsets of $S^{\mathbb{S}}_\theta$, similarly as already discussed in Chapter 5. Before that, the following applies:

$$\begin{aligned}
&\iint\limits_{S^{\mathbb{S}}_\theta} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy\ = \\
&\qquad \iint\limits_{S^{\mathbb{S}}_\theta \cap \mathcal{FS}(S)^{\mathbb{C}}} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy\ + \iint\limits_{S^{\mathbb{S}}_\theta \cap \mathcal{FS}(S)^{\mathbb{S}\setminus\mathbb{C}}} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy \\
&\qquad\quad = Area(S^{\mathbb{S}}_\theta \cap \mathcal{FS}(S)^{\mathbb{C}})\ + \iint\limits_{S^{\mathbb{S}}_\theta \cap \mathcal{FS}(S)^{\mathbb{S}\setminus\mathbb{C}}} \mu_{\mathcal{FS}(S)}(x,y)\ dxdy. \qquad (7.20)
\end{aligned}$$

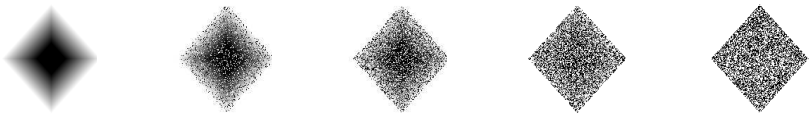
Now, if we consider the partition of $S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S_\theta)^{\mathbb{S}\setminus\mathbb{C}}$ consisting of disjoint subsets

$$\begin{aligned} S_\theta^1 &= \{(x,y) \in S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S_\theta)^{\mathbb{S}\setminus\mathbb{C}} \;\big|\; x<0,\ y\geq 0\}, \\ S_\theta^2 &= \{(x,y) \in S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S_\theta)^{\mathbb{S}\setminus\mathbb{C}} \;\big|\; x<0,\ y<0\}, \\ S_\theta^3 &= \{(x,y) \in S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S_\theta)^{\mathbb{S}\setminus\mathbb{C}} \;\big|\; x\geq 0,\ y\geq 0\}, \\ S_\theta^4 &= \{(x,y) \in S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S_\theta)^{\mathbb{S}\setminus\mathbb{C}} \;\big|\; x\geq 0,\ y<0\}, \end{aligned} \tag{7.21}$$

we can evaluate $\iint\limits_{S_\theta^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\,dxdy$ as follows:

$$\begin{aligned} &\iint\limits_{S_\theta^{\mathbb{S}}} \mu_{\mathcal{FS}(S)}(x,y)\,dxdy = \\ &= Area(S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{\mathbb{C}}) + \iint\limits_{S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{S\setminus C}} \mu_{\mathcal{FS}(S)}(x,y)\,dxdy \\ &= Area(S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{\mathbb{C}}) + \\ &\quad + \frac{1}{a-b}\cdot(-m_{1,0}(S_\theta^1) + m_{0,1}(S_\theta^1) - m_{1,0}(S_\theta^2) - m_{0,1}(S_\theta^2) \\ &\qquad + m_{1,0}(S_\theta^3) + m_{0,1}(S_\theta^3) + m_{1,0}(S_\theta^4) - m_{0,1}(S_\theta^4) \\ &\qquad - b\cdot Area(S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{S\setminus C})) \\ &= Area(S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{\mathbb{S}}) + \\ &\quad + \frac{1}{a-b}\cdot(-m_{1,0}(S_\theta^1) + m_{0,1}(S_\theta^1) - m_{1,0}(S_\theta^2) - m_{0,1}(S_\theta^2) \\ &\qquad + m_{1,0}(S_\theta^3) + m_{0,1}(S_\theta^3) + m_{1,0}(S_\theta^4) - m_{0,1}(S_\theta^4)) \\ &\qquad + \frac{a}{b-a}\cdot Area(S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{S\setminus C}) \end{aligned} \tag{7.22}$$

resulting that, instead of (7.8), it is enough to compute the areas of $S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{\mathbb{S}}$ and $S_\theta^{\mathbb{S}} \cap \mathcal{FS}(S)^{\mathbb{S}\setminus\mathbb{C}}$, and the first order moments of the subsets from (7.21), which further contribute to making the computation of $\mathcal{FS}_q(S)$ faster, but without compromising its simplicity as well.



(a) 1.0000 (b) 0.8725 (c) 0.7525 (d) 0.6280 (e) 0.4928

Figure 7.3: Fuzzy shapes for increasing levels of salt and pepper noise added to a fuzzy square in (a). $\mathcal{FS}_q(S)$ values are given below each shape. Noise levels added to shapes in (b), (c), (d) and (e) are 0.25, 0.5, 0.75 and 1, respectively.

7.3 Experimental illustrations of $\mathcal{FS}_q(S)$

This section deals with many different experiments to provide a better understanding of the new fuzzy squareness measure $\mathcal{FS}_q(S)$. Experiments presented are selected in a way to additionally describe the behaviour of new shape measure as well as to confirm theoretically proven considerations already discussed in the chapter until now. Besides, the experiments comparing the behaviour of the newly derived $\mathcal{FS}_q(S)$ with our initial fuzzy squareness measure $\mathcal{FS}_{fit}(S)$ are included as well.

In the first experiment it considers the behaviour of the new fuzzy squareness measure in the presence of increasing noise levels added to a perfect fuzzy square. Five fuzzy squares, listed following the degree of added salt and pepper noise, are given in Fig. 7.3, along with their measured $\mathcal{FS}_q(S)$ squareness. Results presented are given as the averaged squareness for 100 degraded fuzzy squares at each degree of noise. According to our expectations, the largest $\mathcal{FS}_q(S)$ value, over the shapes displayed in Fig. 7.3, is equal to 1.0000 and corresponds to noise-free fuzzy square (the shape in Fig. 7.3(a)), and then decreases accordingly as the noise level increases. This also stems from our theoretical results of Theorem 7.2, given that a larger added noise contributes to a larger degradation of a perfect fuzzy square. To provide a better illustration of the robustness of $\mathcal{FS}_q(S)$ concerning added noise, a plot of computed $\mathcal{FS}_q(S)$ values for increasing levels of noise is given in Fig. 7.4. Again, plotted squareness values are given as the averaged squareness for all observations at each level of added noise. The noise probability starts from 0, for a perfect fuzzy square in Fig. 7.3(a) and the largest $\mathcal{FS}_q(S)$ equals 1, and then decreases to shape in Fig. 7.3(e) with the minimal measured $\mathcal{FS}_q(S)$ value.

The following experiment illustrates how the new squareness measure $\mathcal{FS}_q(S)$

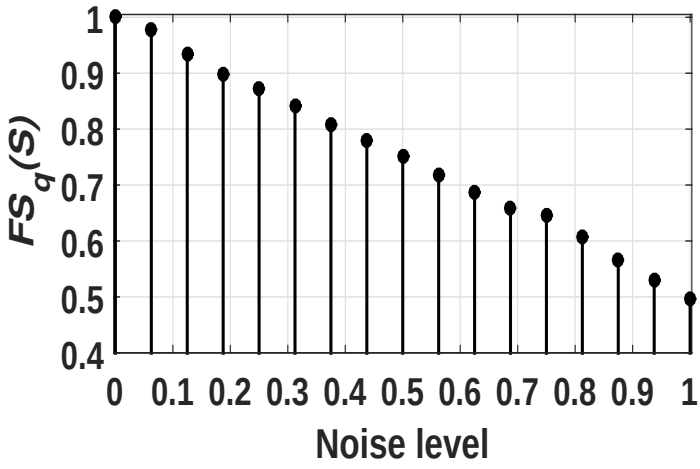


Figure 7.4: Plot of measured $\mathcal{FS}_q(S)$ values for shapes in Fig. 7.3, for increasing levels of salt and pepper noise added to a fuzzy square in Fig. 7.3(a).

behaves as the shape changes under the continuous modification from a fuzzy square to a fuzzy disc, and then to a fuzzy square again. Several such defined fuzzy shapes, given by membership function

$$\mu(x,y;r,R,\varepsilon)=\begin{cases}1 & , \quad |x|^{2/\varepsilon}+|y|^{2/\varepsilon}\leq r^{2/\varepsilon},\\ \dfrac{|x|+|y|-R}{r-R} & , \quad r^{2/\varepsilon}<|x|^{2/\varepsilon}+|y|^{2/\varepsilon}<R^{2/\varepsilon},\\ 0 & , \quad |x|^{2/\varepsilon}+|y|^{2/\varepsilon}\geq R^{2/\varepsilon},\end{cases} \tag{7.23}$$

are presented in Fig. 7.5, together with their measured $\mathcal{FS}_q(S)$ values. It should be noted herein that in the boundary case, for $\varepsilon=0$, the membership function $\mu(x,y;r,R,\varepsilon)$ is by definition reduced to the following form:

$$\mu(x,y;r,R,0)=\begin{cases}1 & , \quad \max\{|x|,|y|\}\leq r,\\ \dfrac{|.x|+|y|-R}{r-R} & , \quad r<\max\{|x|,|y|\}<R,\\ 0 & , \quad \max\{|x|,|y|\}\geq R.\end{cases} \tag{7.24}$$

Thus the shapes presented in Fig. 7.5 modify from a fuzzy square ($\varepsilon=0$) to a fuzzy disc ($\varepsilon=1$), and then again to a fuzzy square ($\varepsilon=2$). It can be noticed that

(a) 1.0000 (b) 0.9954 (c) 0.9814 (d) 0.9950 (e) 1.0000

Figure 7.5: Five shapes defined by (7.23) for $\varepsilon \in \{0, 0.5, 1, 1.5, 2\}$, together with their assigned $\mathcal{FS}_q(S)$ values.

the results presented are in accordance with human perception, since, as the fuzzy shape becomes more circular, the measured fuzzy squareness decreases from a perfect fuzzy square (the shape in Fig. 7.5(a)), through the rounded fuzzy square (given in Fig. 7.5(b)) to a fuzzy disc given in Fig. 7.5(c). On the other side, the measured fuzzy squareness increases as the fuzzy shapes become more fuzzy square (see, for example, shapes in Fig. 7.5(d) and 7.5(e) relative to a shape given in Fig. 7.5(c) (i.e., fuzzy disc)). As expected, the largest $\mathcal{FS}_q(S)$ value (among the shapes presented in Fig. 7.5) corresponds to shapes in Fig. 7.5(a) and 7.5(e) (i.e., for perfect fuzzy squares), while the smallest squareness equal to 0.9814 is reached for a shape of fuzzy disc presented in Fig. 7.5(c), which can be considered to be least fuzzy square, among the shapes in Fig. 7.5. These results are also consistent with our theoretical results, since, as already proven, the fuzzy squareness measure $\mathcal{FS}_q(S)$ assigns the value 1 if and only if the shape given is a fuzzy square. For a better understanding of the behaviour of $\mathcal{FS}_q(S)$, Fig. 7.6 presents a plot of measured $\mathcal{FS}_q(S)$ values for increasing values of parameter ε. The plot decreases from the maximal squareness 1.0000, for a perfect fuzzy square in Fig. 7.5(a), to 0.9814, assigned to a fuzzy disc in Fig. 7.5(c), representing its minimal value as well, and then increases to the largest squareness 1.0000, corresponding to a fuzzy square in Fig. 7.5(e). It can be noticed that all the computed $\mathcal{FS}_q(S)$ values are also following the theoretical observations that the new measure is invariant to similarity transformations. This follows from the fact that the second half of the shapes in Fig. 7.5 can be obtained through the corresponding similarity transformations (e.g., rotation and scaling) from the shapes given in the first half of the same figure.

Now, we observe several synthetic shapes to illustrate how the new fuzzy squareness measure $\mathcal{FS}_q(S)$ acts in comparison with $\mathcal{FS}_{fit}(S)$. The first six fuzzy shapes

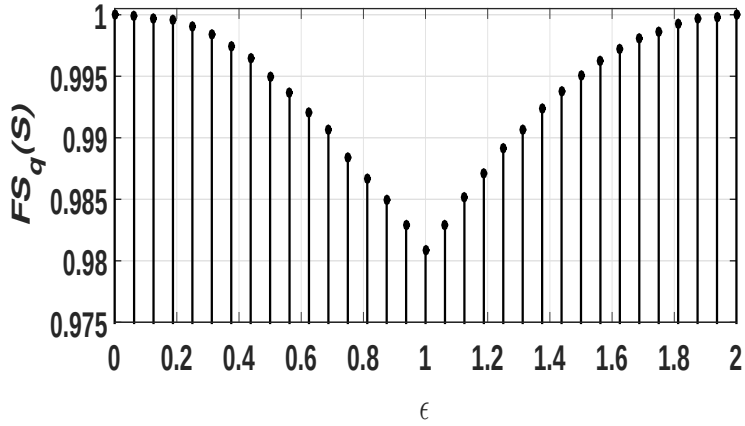


Figure 7.6: Plot of measured $\mathcal{FS}_q(S)$ values for shapes, defined in (7.23), for increasing values of parameter ε.

in Fig. 7.1 demonstrate how the squareness measures act on shapes with holes. The shapes presented correspond to fuzzy star shapes differing by the relative size, position and number of the holes within the shape, arranged according to decreasing $\mathcal{FS}_q(S)$ and $\mathcal{FS}_{fit}(S)$ values. It can be noticed that, as the size of the holes increases (e.g., the second and third shapes in Fig. 7.1), then the measured squareness decreases. To illustrate how the considered fuzzy shape measures depend on the number of holes, we observe the following three fuzzy shapes in the same figure. As expected, with the increased number of the holes inside the shape, the measured squareness decreases. Besides, the largest measured $\mathcal{FS}_q(S)$ and $\mathcal{FS}_{fit}(S)$ values are assigned to fuzzy star shape without holes (i.e., the first shape in Fig. 7.1). These results also match our perception about the behaviour of the fuzzy squareness measures since we prefer that the fuzzy squareness decreases as the size of the holes and their relative number increases from the left to the right. This is also in a line with our initial request that the fuzzy squareness shape measures should evaluate to what extent a given fuzzy shape matches an ideal fuzzy square. It can also be noticed that both squareness measures provide the same ordering among the shapes considered, which has been already proven to be of particular benefit in some applications.

Besides, it can be noticed that in the case of the last four shapes in Fig. 7.1 there exist shapes for which the fuzzy squareness $\mathcal{FS}_{fit}(S)$ is not able to provide a clear distinction among them. The shapes presented are generated from the circle line (the seventh shape in Fig. 7.1) by excluding its quarters. The changes in the measured fuzzy squareness $\mathcal{FS}_q(S)$ are following our perception, which can be understood to

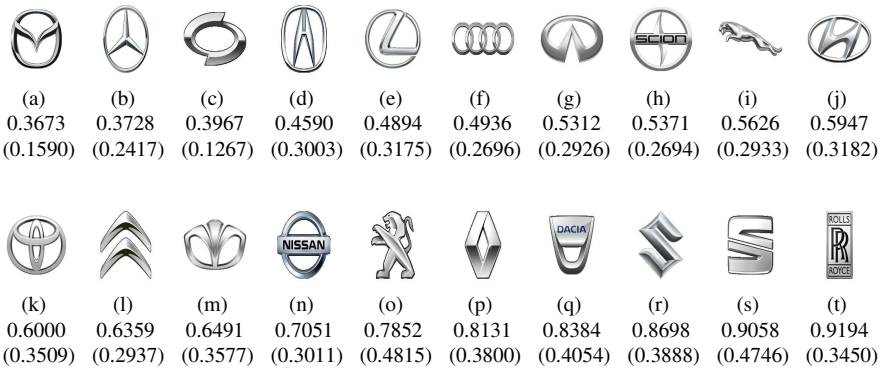
be an advantage compared to $\mathcal{FS}_{fit}(S)$. This follows from the fact that most fuzzy square shape is the last shape with assigned $\mathcal{FS}_q(S)$ value equal to 0.0170, while least fuzzy square shape is a circle line (the seventh shape in Fig. 7.1) with measured $\mathcal{FS}_q(S)$ equal to 0.0011. As it can be seen, the measured fuzzy squareness $\mathcal{FS}_{fit}(S)$ for the last four shapes is equal to 0. Such a property of $\mathcal{FS}_{fit}(S)$ is considered, as already mentioned, to be a disadvantage in relation to $\mathcal{FS}_q(S)$. However, as already indicated in the case of hexagonality measures from Chapter 5, such a correspondence established using two similar measures, which evaluate the same property of the shape under consideration, can be particularly important in some applications, since, as such, they can be combined to improve performance in a variety of object analysis tasks. In this regard, we illustrate how the new fuzzy squareness measure corresponds to human perception. Following this, it is applied to several randomly selected image data from several commonly used image databases, whereas the generality of the new fuzzy measures is illustrated by their applicability to different types of image data. Let mention some of them:

1. For a binary image, the city block DT (Definition 3.4) is applied to an inverse of the convex hull for a given object is computed. Then the new fuzzy squareness measure is applied to the normalized fuzzy set whose support coincides with the original foreground object.

2. For a given gray-level image, the foreground objects can be viewed as the corresponding fuzzy shapes to which the new fuzzy shape measure can be applied (method 2.1). Also, a gray-level image can be binarized using the Otsu thresholding method [107], and then the new fuzzy squareness measure is applied according to the method described in 1 (method 2.2).

3. For a given color image, its transformation into a gray-level image is first performed, given that the color does not play a role in our considered features. The new fuzzy squareness shape measure is then applied following the methods described in 2.

Figure 7.7 illustrates a large variety of randomly selected automotive logos taken from [1] that are presented as gray-level images (according to methods 3 and 2.1). The shapes are listed according to their increasing $\mathcal{FS}_q(S)$ values, while the measured $\mathcal{FS}_{fit}(S)$ values are given below in the brackets. It can be said that the obtained order is not in contradiction with our initial request given the fuzzy squareness quantifies how much a given shape differs from the fitted fuzzy square. If the shapes are ranked according to $\mathcal{FS}_{fit}(S)$, then the ordering (a) (b) (c) (d) (e) (f) (g) (h) (i) (j) (k) (l) (m) (n) (o) (p) (q) (r) (s) (t) established by $\mathcal{FS}_q(S)$ is replaced with (c) (a) (b) (h) (f) (g) (i) (l) (d) (n) (e) (j) (t) (k) (m) (p) (r) (q) (s) (o). It can be noticed that the biggest differences are in the case of highest ranked shapes by $\mathcal{FS}_q(S)$, which is ranked in the mean if the fuzzy squareness $\mathcal{FS}_{fit}(S)$ is applied. Such behaviour of



(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)
0.3673	0.3728	0.3967	0.4590	0.4894	0.4936	0.5312	0.5371	0.5626	0.5947
(0.1590)	(0.2417)	(0.1267)	(0.3003)	(0.3175)	(0.2696)	(0.2926)	(0.2694)	(0.2933)	(0.3182)

(k)	(l)	(m)	(n)	(o)	(p)	(q)	(r)	(s)	(t)
0.6000	0.6359	0.6491	0.7051	0.7852	0.8131	0.8384	0.8698	0.9058	0.9194
(0.3509)	(0.2937)	(0.3577)	(0.3011)	(0.4815)	(0.3800)	(0.4054)	(0.3888)	(0.4746)	(0.3450)

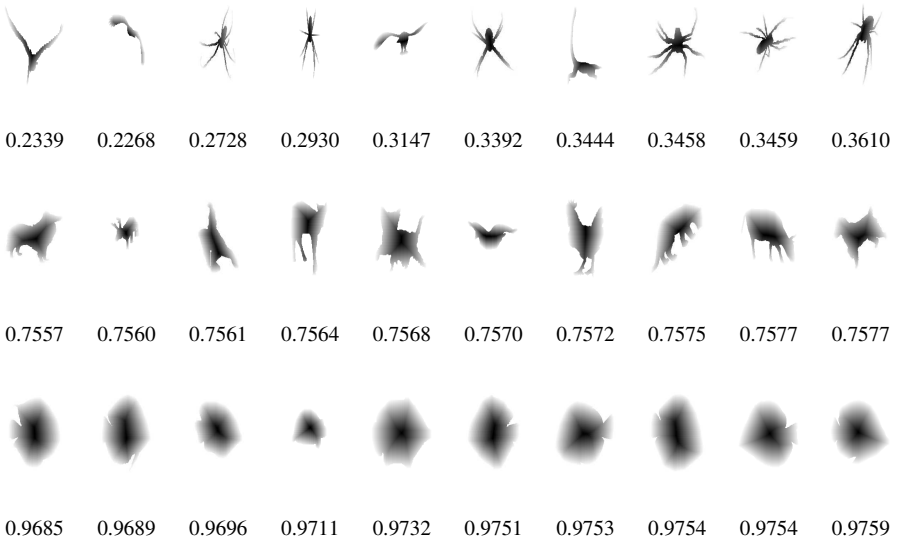
Figure 7.7: Shapes of automotive logos along with their assigned $\mathcal{FS}_q(S)$ and $\mathcal{FS}_{fit}(S)$ values. The computed $\mathcal{FS}_{fit}(S)$ values are in the brackets.

$\mathcal{FS}_q(S)$ can be understood as an advantage over $\mathcal{FS}_{fit}(S)$, given that, according to human perception, a higher squareness should receive the shape present in Fig. 7.7(t) instead of, for example, the shapes given in Fig. 7.7(k) and 7.7(m). As expected, the smallest measured $\mathcal{FS}_{fit}(S)$ value is assigned to the shape in Fig. 7.7(c) since such a shape has the largest portion of non-foreground pixels. Besides, we can also notice that the rankings according to both squareness measures differ significantly for the shape given in Fig. 7.7(o). In fact, highly ranked shapes in Fig. 7.7(p) and (s), according to $\mathcal{FS}_q(S)$, are ranked as lower fuzzy square shapes than the shape in Fig. 7.7(o). Such a property of $\mathcal{FS}_{fit}(S)$ can be viewed as its weakness, since the shapes in Fig. 7.7(p) and (s) are recognized to be more fuzzy squared than the shape in Fig. 7.7(o). Also, it can be noticed that the highly ranked shapes in Fig. 7.7 represent more compact shapes containing the larger portions of the object pixels. These results match with human perception, given that as most compact shape, among the shapes presented, can be considered the shape with the largest measured squareness $\mathcal{FS}_q(S)$ equal to 0.9194. On the other side, the shapes having larger proportions of long and elongated thin features, or containing multiple diverse holes are tending to be perceived as lower-ranked shapes according to fuzzy squareness measures. The results obtained are following such a perception, given that the fuzzy squareness measures $\mathcal{FS}_q(S)$ and $\mathcal{FS}_{fit}(S)$ assign lower squareness for the shapes given in the first row in Fig. 7.7.

Examples of several logos of the different tire brands from [2] are presented in Fig. 7.8 (previously prepared following the above-described methods 3 and 2.2). The same reasoning applies as in the previous figure, since the lower $\mathcal{FS}_q(S)$ values are assigned to the shapes containing larger areas of non-foreground pixels (e.g., the

(a)	(b)	(c)	(d)	(e)	(f)	(g)
0.4540	0.6528	0.6651	0.7037	0.7301	0.8437	0.9172
(0.4201)	(0.4208)	(0.5227)	(0.5015)	(0.6101)	(0.6672)	(0.7592)

Figure 7.8: Examples of logos of tire brands, and their assigned $\mathcal{FS}_q(S)$ values arranged in ascending order. The measured $\mathcal{FS}_{fit}(S)$ are in the brackets.



0.2339	0.2268	0.2728	0.2930	0.3147	0.3392	0.3444	0.3458	0.3459	0.3610
0.7557	0.7560	0.7561	0.7564	0.7568	0.7570	0.7572	0.7575	0.7577	0.7577
0.9685	0.9689	0.9696	0.9711	0.9732	0.9751	0.9753	0.9754	0.9754	0.9759

Figure 7.9: From the top to the bottom: the ten lowest, intermediate and highest-ranked shapes, according to increasing $\mathcal{FS}_q(S)$ values, from the animal dataset [10]. $\mathcal{FS}_q(S)$ values are given below the corresponding shapes.

first three shapes in Fig. 7.8) than the shapes containing a larger fraction of object pixels (e.g., the remaining shapes in the same figure). If the same shapes are ranked concerning fuzzy squareness $\mathcal{FS}_{fit}(S)$ (given in the brackets), a slightly different ranking is then achieved. In fact, the third and fourth shapes of tire logos should change their places if the ranking according to increasing $\mathcal{FS}_{fit}(S)$ is applied. It can be said that such a ranking is in contradiction with our expectations since we

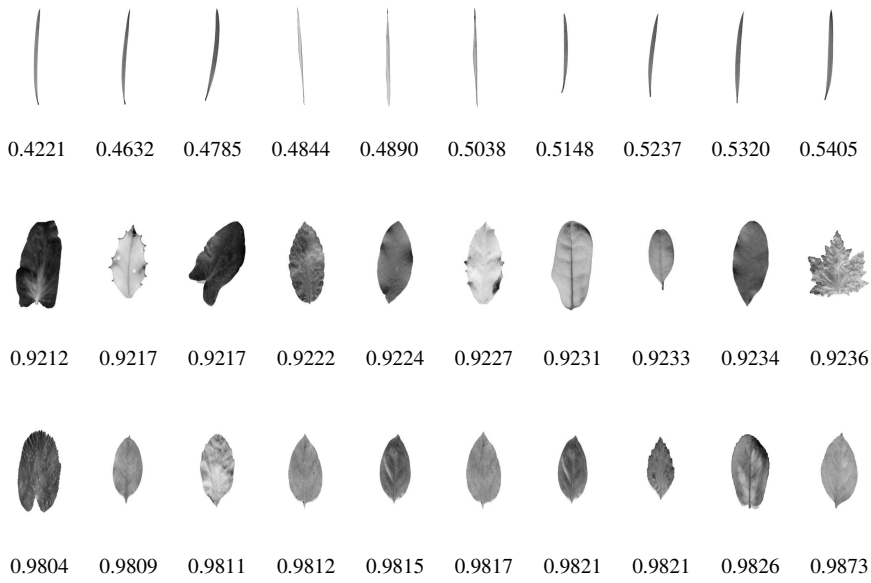


Figure 7.10: The ten lowest, intermediate and highest arranged shapes according to $\mathcal{FS}_q(S)$ (given below each shape related) from the Portuguese Leaves database [141].

expect a higher squareness assigned to the fourth rather than to the third logo shape. Again, these results are consistent with the theoretical results, given that, according to Definition 7.10, a new fuzzy squareness $\mathcal{FS}_q(S)$ defines the degree to which the measured shape is similar to its associated fuzzy square $\mathcal{FS}(S)$, as defined in (7.6).

In the following, we will illustrate the behaviour of a new fuzzy squareness on several known and commonly studied image datasets. The first dataset is the animal dataset [10], already discussed in Chapter 5. Figure 7.9 presents the ten lowest and highest ranked shapes according to their increasing $\mathcal{FS}_q(S)$ squareness, together with ten intermediate shapes. The presented animal shapes are given as fuzzified version of the original binary shapes, following method 1. It can be noticed that ranking matches our expectation since $\mathcal{FS}_q(S)$ evaluates the degree of how much a given fuzzy shape differs from fitted fuzzy square $\mathcal{FS}(S)$. Following this, it is easy to understand why the shapes with thin and elongated features such as wings, tails, or legs are assigned lower $\mathcal{FS}_q(S)$ values, while a higher squareness is assigned to more compact shapes containing a larger portion of the foreground pixels (e.g., the shapes in the second and third rows). Besides, we can note that by comparing the obtained order with the one established using a new measure of shape hexagonality, as presented in Chapter 5, the results achieved are following what is expected. This follows from the fact that both

introduced shape measures are based on the area of the shape considered, given that they are defined utilizing the corresponding integrals over the whole support of the shape). This is the reason why such established orders are in agreement concerning our perception of their behaviour.

The second well-known database is Portuguese Leaves dataset [141], containing a total of 340 leaf images[3]. The ten lowest, intermediate and highest-ranked leaf shapes according to new fuzzy squareness $\mathcal{FS}_q(S)$ are in Fig. 7.10. The shapes are given as gray-level images prepared according to the aforementioned method 2.1. It can be noticed that lower scores of the measured squareness correspond to the shapes recognized as more linear and elongated than the ones presented in Fig. 7.10. These results do support our theoretical results proven in the previous section, and which have been already discussed in some of the previous experiments. Also, it can be concluded again that the ten highest-ranked shapes are perceived to be most compact shapes in the figure, containing a larger fraction of the foreground pixels.

The third considered dataset, MPEG-7 CE-1 from [73] that has already been observed in Chapter 5 to evaluate the performance of the new measures of hexagonality as well as their associated by-products. Contrary to these experiments, we now illustrate how the new fuzzy squareness measures applied to the appropriate fuzzified examples of the initial corresponded shapes. Examples of ten lowest, middle and highest-ranked shapes according to measured $\mathcal{FS}_q(S)$ values belonging to this dataset, previously prepared to applying fuzzy squareness measures (following the method 1, given above) are in Fig. 7.11. Again, it can be noted that the shapes with long thin and/or elongated object areas, or characterized by a larger fraction of non-object features are identified by $\mathcal{FS}_q(S)$ as lower fuzzy square shapes. Regarding the ten highest-ranked shapes, it is easy to notice that they are the squares with different degree of shape deformations, which can be perceived as most square shapes, among the shapes presented. The results obtained do not contradict our theoretical observations relating to the behaviour of the new fuzzy squareness measure. This explains why the shapes represented as most compact fuzzy shapes, among the presented shapes, are assigned a higher fuzzy squareness $\mathcal{FS}_q(S)$. Finally, it is interesting to provide a comparison of these results with those obtained in Chapter 5 when shapes from the same image dataset are ranked according to measured hexagonality. Regarding the shapes given in the first two rows of Fig. 5.9 and Fig. 7.11, we can notice that the obtained results are in line with our expectation, taking into account the definitions and properties of these measures, as well as observation already discussed in the previous experiments. However, the most interesting are the shapes given in the third row of the corresponding figures. As the new hexagonality measure

[3]Although the presented database contains 40 different plant species, the attached zip file which is provided by the authors contains a total of 340 leaf images (classified into 30 different species), which were recognized by the authors to be simple leaves images. Both types of images, RGB and binary, are provided in the file attached.

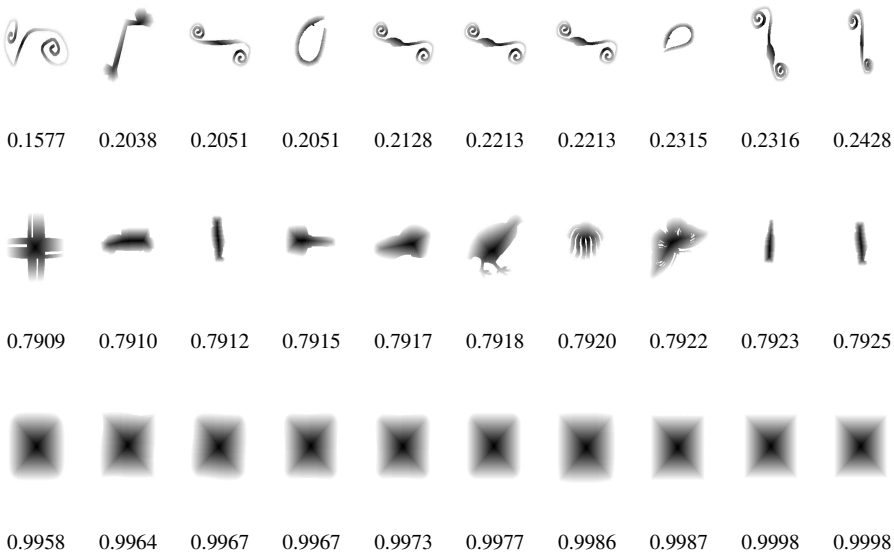


Figure 7.11: The first, second and third row present the ten lowest, middle and highest fuzzy square shapes from MPEG-7 CE-1 [73], according to assigned $\mathcal{FS}_q(S)$ values, respectively. The shapes presented are given as the corresponding fuzzified shapes according to method 1, described in the text.

classifies the squares with different degrees of deformation and corresponding pencil shapes, as most hexagonal ones among the shapes presented, on the other hand, a new fuzzy squareness measure classifies the fuzzified versions of the corresponding squares given in the original image dataset as most fuzzy square. This is again in line with our initial requirements regarding the behaviour of these two measures, given that the new measure of hexagonality in a degenerative case can recognize square as most hexagonal shape as well. Besides, the different order of the observed shapes concerning these two considered measures can be of particular benefit and importance in the different tasks of analyzing and comparing objects based on their shape, which will be detailed and discussed in the following section.

7.4 Experiments illustrating the applicability of fuzzy squareness

This section illustrates the effectiveness of the new fuzzy squareness measures in the tasks of classifying images from several modern image datasets. Also, our attention will be directed to the application of simpler shape-based analysis tools, relatively simple and easily computable descriptors of the object, well-founded classification algorithms, as well as the applied classifier. Such aspects will be of particular importance in experiments not aimed to achieve the benchmark result regarding the image dataset considered, nor to apply potentially complex tools which are known to improve efficiency in selected tasks of object analysis. For example, in some presented classification tasks, which are applied to two well-known Leaf datasets from [151] and [141], we will present the experiments involving a small number of relatively simple descriptors to obtain the results which are comparable to those presented in the literature up to date.

First classification task. First classification experiment illustrates that the new fuzzy measures can be used together with several simple shape features to increase the efficiency of classifying images from MPEG-7 CE-1 dataset [73]. We have implemented the nearest neighbors classification algorithm with Mahalanobis distance. Herein we present two groups of the experiments differing according to initially chosen descriptors. The achieved leave-one-out efficiencies are shown in Table 7.1. Particular emphasis will be given to the comparison of the results obtained with those achieved in Chapter 5 when the new hexagonality measures and their side-results are included into consideration.

The first group of experiments starts with the area-based and perimeter-based convexities[2], the multi-component shape disconnectedness measure $\mathcal{D}(S)$ [162], along with hexagonality measures $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ from [69]. Such selected shape features reach a leave-one-out classification accuracy of 79.00%. If it is augmented with initial fuzzy squareness shape measure $\mathcal{FS}_{fit}(S)$, the classification efficiency has improved to 82.79%. Better accuracy is achieved by adding the new fuzzy squareness $\mathcal{FS}_q(S)$ instead: 83.50%. A further improvement has been provided combining both $\mathcal{FS}_{fit}(S)$ and $\mathcal{FS}_q(S)$ together with five initial shape descriptors. The classification performance is then increased to 86.14%, thus providing an improvement higher than 7%. Such an improvement in accuracy is approximately equal to the one achieved in Chapter 5, using one descriptor less (i.e., a total of seven simple shape descriptors). Also, this relatively high result is better than the results achieved in the following two groups of experiments performed on this database using no more than nine different descriptors of shape. For more details, see results in Table 5.1. Also, performed experiments show that new fuzzy squareness measures are compatible, and can be combined to achieve a better performance in such a task of classification. This is also illustrated if only $\mathcal{FS}_{fit}(S)$ and $\mathcal{FS}_q(S)$ are applied separately or together, without

Table 7.1: Leave-one-out classification accuracies obtained using the corresponding shape descriptors. Improvements achieved by including the new fuzzy shape measures are displayed as bolded.

	Shape features	Rates
First group	5 shape features	79.00%
	5 shape features + $\mathcal{FS}_{fit}(S)$	**82.79**%
	5 shape features + $\mathcal{FS}_{q}(S)$	**83.50**%
	5 shape features + $\mathcal{FS}_{q}(S)$ + $\mathcal{FS}_{fit}(S)$	**86.14**%
Second group	7 shape features	83.29%
	7 shape features + $\mathcal{FS}_{q}(S)$	**86.29**%
	7 shape features + $\mathcal{FS}_{fit}(S)$	**86.64**%
	7 shape features + $\mathcal{FS}_{fit}(S)$ + $\mathcal{FS}_{q}(S)$	**88.93**%

initial five shape measures. Thus, for example, if $\mathcal{FS}_{fit}(S)$ and $\mathcal{FS}_{q}(S)$ are considered separately then the achieved classification scores are of 22.57% and 27.29%, respectively, while a better score of 51.43% is obtained if they are combined.

In the second group of experiments, three global shape descriptors such as rectangularity, roundness and compactness are combined together with area and perimeter-based convexities[2] and hexagonality measures $\mathcal{H}(S)$ and $\mathcal{H}_{fit}(S)$ from Chapter 5, instead of $\mathcal{D}(S)$. Such combined descriptors achieve a classification rate of 83.29%. If the new fuzzy squareness $\mathcal{FS}_{q}(S)$ is added into consideration, then the achieved efficiency increases to 86.29%, whereas including the fuzzy squareness measure $\mathcal{FS}_{fit}(S)$ instead, slightly better accuracy of 86.64% has been reached. If both fuzzy squareness measures $\mathcal{FS}_{fit}(S)$ and $\mathcal{FS}_{q}(S)$ are added into consideration, the classification accuracy increases further to 88.93%. It can be noticed that such a pretty good result represents as well a relative high improvement of nearly 6% compared to the initial five descriptors described in this group of experiments. This accuracy also outperforms the highest achieved efficiency in the previous group of experiments, and also the best achieved in Chapter 5 of nearly 88%, using a total of 9 simple shape descriptors (one descriptor less than in Chapter 5). Moreover, such a good result of approximately 89% represents also the best classification result (i.e., benchmark result) for this image database. Notice that the results presented illustrate that both introduced hexagonality measures from Chapter 5 together with both fuzzy squareness shape measures include mostly independent shape information, which further contributes to keeping the classification efficiency at a relatively high rate. Besides, according to the results given in Table 7.1, it is easy to notice that there are several

more combinations of the shape features that can be combined to increase the efficiency, and which are also comparable to those achieved in the literature to date. More about this can be found in Chapter 5.

Second classification task. This experiment demonstrates the advantages of the new fuzzy squareness measures in the task of classifying leaf images from the Swedish Leaf dataset [151]. As already discussed, this dataset contains 1125 color images (15 classes, 75 images per class). Examples of leaf images taken from each class are in Fig. 5.12. This classification task represents a well-known image analysis task, given high similarities among the leaf images belonging to different classes. It should be noted at the beginning of the experiment that the emphasis here does not refer to obtaining the best possible result, i.e., benchmark result, but to presenting a relatively simple approach that relies on the use of several easily computable shape descriptors applied together with our new fuzzy squareness measures to illustrate their efficiency in this classification task. The initial set of shape descriptors now consists of multi-component shape measure $\mathcal{D}(S)$ [162], multi-component shape measures from [115] (denoted as $Imcc(S)$ and $IImcc(S)$), and the first Hu moment invariant [66]. The classification results achieved using the nearest neighbors machine learning-based classifier are given in Table 7.2. Such a defined set of shape descriptors produces a leave-one-out classification rate of 79.26%. By including two versions of the fuzzy squareness shape measure $\mathcal{FS}^1_{st}(S)$ and $\mathcal{FS}^2_{st}(S)$[4] separately, the classification score has increased to 81.48% and 82.52%, respectively. Greater improvements in the accuracy of 86.07% and 92.44% have been achieved, respectively, by including two versions of the new squareness measure $\mathcal{FS}^1_q(S)$ and $\mathcal{FS}^2_q(S)$, instead. A further improvement of 93.33% has been achieved combining both versions $\mathcal{FS}^1_q(S)$ and $\mathcal{FS}^2_q(S)$, while the additional improvement of 94.67% is obtained by adding the first version of $\mathcal{FS}^1_q(S)$ together with the second version of the fuzzy squareness measure $\mathcal{FS}^2_{st}(S)$. Such a great relatively high improvement of nearly 15.5% represents a pretty good classification result, outperforming most achieved rates in the literature to date, given that only six simple descriptors have been used. For example, it is worth noting that such a classification result outperforms the efficiency of 89.1%, achieved in [115] using only seven Hu moment invariants, as well as the efficiency of 94.9% based on the use of seven multi-component moment invariants introduced in [115].

Also, the experiments presented illustrate that the new measures are complementary, and include independent information of the shape. This can be further illustrated by considering the results achieved using only fuzzy squareness measures without the initial five descriptors. The classification results are given in the second part of Table 7.2. As noticed in Table 7.2, both versions of the fuzzy squareness measures

[4]The first version of the fuzzy squareness measure is applied to gray-level leaf image viewed as a fuzzy shape (following the method 2.1, given at the beginning of the previous subsection), while the second version is applied to the distance map of the binarized leaf image (method 2.2 in the same subsection)

Table 7.2: Accuracies obtained by combining initial descriptors with the new fuzzy shape measures. Results obtained by including the new shape measures are given as bold.

Shape features	Rates
4 shape features	79.26%
4 shape features + $\mathcal{FS}^1_{st}(S)$	**81.48%**
4 shape features + $\mathcal{FS}^2_{st}(S)$	**82.52%**
4 shape features + $\mathcal{FS}^1_{q}(S)$	**86.07%**
4 shape features + $\mathcal{FS}^2_{q}(S)$	**92.44%**
4 shape features + $\mathcal{FS}^1_{q}(S) + \mathcal{FS}^2_{q}(S)$	**93.33%**
4 shape features + $\mathcal{FS}^1_{q}(S) + \mathcal{FS}^2_{st}(S)$	**94.67%**
$\mathcal{FS}^1_{q}(S)$	**42.67%**
$\mathcal{FS}^1_{st}(S)$	**48.00%**
$\mathcal{FS}^2_{st}(S)$	**54.67%**
$\mathcal{FS}^2_{q}(S)$	**56.00%**
$\mathcal{FS}^1_{q}(S) + \mathcal{FS}^1_{st}(S)$	**62.37%**
$\mathcal{FS}^2_{q}(S) + \mathcal{FS}^2_{st}(S)$	**64.89%**
$\mathcal{FS}^1_{q}(S) + \mathcal{FS}^2_{st}(S)$	**67.70%**
$\mathcal{FS}^1_{q}(S) + \mathcal{FS}^2_{q}(S)$	**68.15%**
$\mathcal{FS}^1_{st}(S) + \mathcal{FS}^2_{q}(S)$	**73.04%**
$\mathcal{FS}^1_{q}(S) + \mathcal{FS}^2_{q}(S)$	**77.19%**
$\mathcal{FS}^1_{q}(S) + \mathcal{FS}^2_{q}(S) + \mathcal{FS}^1_{st}(S)$	**77.48%**
$\mathcal{FS}^1_{q}(S) + \mathcal{FS}^2_{q}(S) + \mathcal{FS}^2_{st}(S)$	**82.81%**
$\mathcal{FS}^1_{q}(S) + \mathcal{FS}^1_{st}(S) + \mathcal{FS}^2_{q}(S) + \mathcal{FS}^2_{st}(S)$	**86.96%**

are relatively efficient in this classification task. A leave-one-out accuracies obtained using separately $\mathcal{FS}^1_{q}(S)$, $\mathcal{FS}^1_{st}(S)$, $\mathcal{FS}^2_{st}(S)$ and $\mathcal{FS}^2_{q}(S)$ are of 42.67%, 48.00%, 54.67% and 56.00%, respectively. As expected, better efficiencies are reached by appropriate combining two by two versions of the new fuzzy measures. For example, the classification accuracy has improved to 62.37% if $\mathcal{FS}^1_{q}(S)$ and $\mathcal{FS}^1_{st}(S)$ are used together; 64.89% if $\mathcal{FS}^2_{q}(S)$ and $\mathcal{FS}^2_{st}(S)$ are used, instead; 67.70% for $\mathcal{FS}^1_{q}(S)$ and $\mathcal{FS}^2_{st}(S)$; 68.15% for $\mathcal{FS}^1_{q}(S)$ and $\mathcal{FS}^2_{q}(S)$; 73.04% for $\mathcal{FS}^1_{st}(S)$ and $\mathcal{FS}^2_{q}(S)$; and

77.19% for $\mathcal{FS}_q^1(S)$ and $\mathcal{FS}_q^2(S)$. Further improvements in the accuracy of 77.48% and 82.81% are then achieved combining both versions of $\mathcal{FS}_q(S)$ with the first and second version of $\mathcal{FS}_{fit}(S)$, respectively. Additional improvement of 86.96% has been achieved if both versions of $\mathcal{FS}_q(S)$ and $\mathcal{FS}_{fit}(S)$ are employed together to improve the classification performance. It can also be noticed that such an achieved accuracy represents better result than those achieved in the first four experiments using five shape descriptors (i.e., one descriptor more). At the end of the experiment, it is worth noting that the results obtained are also comparable with those already existing methods specially designed for this image database. These methods usually involve a much larger set of descriptors, a feature vector space of larger dimensionality, or maybe a more complex classifier. More about these methods have been already said in Chapter 5.

Third classification task. This experiment presents how the new fuzzy squareness measures can be efficiently applied in the task of classifying the leaf images from the Portugues Leaf dataset [141]. As already mentioned, the database considered contains a total of 340 leaf images, classified into 30 different plant species. Several examples of leaf images belonging to this database are given in Fig. 7.10. This widely studied image database was the object of research of several different authors aimed to improve the performance of various classification algorithms. In [141] the authors studied a new shape-based analysis method using the Distance Transform Network (DTN) which combines both complex network (CN) and Euclidean distance transform (EDT). Such a designed method has proved to be particularly effective in the classification of images belonging to this database, giving the efficiency of 85.00% based on the Linear Discriminant Analysis (LDA) classifier, as well as of 77.79% using the Support Vector Machine. In addition, this method also provides a pretty high relative improvement in accuracy of approximately 13% using LDA classifier, or 12% for SVM classifier, with respect to the second-highest efficiencies of nearly 72% and 66% achieved using the complex network (CN) degree algorithm [9] (as reported in [141]). The authors also listed several results achieved for this database. Among them are: 72.35% and 63.47% using 34 features obtained from segment analysis [125] based on LDA and SVM classifier, respectively; 65.75% and 64.41% obtained using 25 curvature descriptors [170] with SVM and LDA classifier, respectively; 53.52% and 63.82% using multiscale fractal dimension and contour saliences [37] based on SVM and LDA classifier, respectively, with 50 applied shape features; etc.

In our experiment, we rely on the approach consisting of several already mentioned shape descriptors strengthened by new fuzzy squareness shape measures. Table 7.3 presents the achieved classification rates. The initial set of shape features includes rectangularity and area-based convexity, multi-component shape measures $\mathcal{D}(S)$ and $Imcc(S)$ from [115], hexagonality based elongation introduced in Chapter 5 as well as the ellipticity measure (denoted as $\mathcal{E}(S)$ in [160]). The nearest neighbors

Table 7.3: Efficiencies achieved using the initial shape descriptors along with the new fuzzy shape measures, performed on the Portuguese Leaves dataset [141]. Results obtained including the new shape measures are given in bold.

Shape features	Rates
6 shape features	63.53%
6 shape features + $\mathcal{FS}_q^1(S)$	**67.65**%
6 shape features + $\mathcal{FS}_q^2(S)$	**68.24**%
6 shape features + $\mathcal{FS}_{st}^2(S)$	**70.29**%
6 shape features + $\mathcal{FS}_q^1(S)$ + $\mathcal{FS}_q^2(S)$	**72.35**%
6 shape features + $\mathcal{FS}_q^1(S)$ + $\mathcal{FS}_q^2(S)$ + $\mathcal{FS}_{st}^2(S)$	**75.00**%
$\mathcal{FS}_q^1(S)$	**32.06**%
$\mathcal{FS}_q^1(S)$ + $\mathcal{FS}_{st}^1(S)$	**38.82**%
$\mathcal{FS}_q^1(S)$ + $\mathcal{FS}_q^2(S)$	**40.59**%
$\mathcal{FS}_q^2(S)$ + $\mathcal{FS}_{st}^2(S)$	**54.71**%
$\mathcal{FS}_q^1(S)$ + $\mathcal{FS}_{st}^1(S)$ + $\mathcal{FS}_{st}^2(S)$	**55.88**%

classifier provides a leave-one-out validation score of 63.53%. This result can be further improved to 67.65%, 68.24% and 70.29% by including separately the first and second version of $\mathcal{FS}_q(S)$, as well as the second version of $\mathcal{FS}_{fit}(S)$[4], respectively. Better result of 72.35% was obtained if both versions of fuzzy squareness $\mathcal{FS}_q(S)$ have been combined with the initial set of shape features. Further improvement of 75.00% was achieved if the second version of $\mathcal{FS}_{fit}(S)$ was included into consideration. Such a pretty good classification score, achieved using only nine simple shape features, also represents a big relative improvement in accuracy of approximately 11.5%, as a result of including our new fuzzy squareness shape measures. As views in Table 7.3, it can be said that both versions of the new squareness measures are of particular importance in this classification task, given that they involve mostly independent information about shapes. This can be further demonstrated if only the fuzzy shape measures are used without initial shape descriptors. For example, if only the first version of $\mathcal{FS}_q(S)$ is applied then the achieved efficiency is of 32.06%, while if both first version of $\mathcal{FS}_{fit}(S)$ and second version of $\mathcal{FS}_q(S)$ are added separately then the accuracy has improved to 40.59% and 54.71%, respectively. Better result for such a dataset of 55.86% was obtained if only the first version of $\mathcal{FS}_q(S)$ and both versions of $\mathcal{FS}_{fit}(S)$ are combined. It can also be noticed that such results are competitive to those already mentioned (as listed in [116]) using the feature vectors

of higher dimensionality. On the other side, our considered approach is based on the use of several simple and easy to compute shape measures, making it interesting from the aspect of applicability and simplicity in a variety of shape-based object analysis tasks.

Chapter 8

Concluding remarks and further research

This chapter concludes the thesis by providing a brief overview of the research and original contributions presented in the thesis. The original contributions and results of the thesis can be naturally divided into two groups, according to the applied methodologies of obtaining scientific results that were followed during the research performed. All the contributions, as well as their mutual connections and overlaps, are clearly illustrated and discussed. The first group of research results presented in the thesis, concretely in Chapters 4 and 6 deals with studying the new methods for estimating three-dimensional Euclidean distance transform (3D EDT), as well as the signature of a two-dimensional shape utilizing the information available from the coverage image representation. In this image representation, the intensity value assigned to each image element is proportional to its relative coverage by a continuous object. We have shown how such image information can be utilized to provide a precise estimate of the position of the continuous object within an image element. This sub-element estimate of the object position has been proven to provide an improvement of the aforementioned shape-based image processing techniques. Improvements achieved include reduced bias and variance, increased rotational and translational invariance, along with increased robustness to the presence of noise that is irresistibly present in the image due to the imperfection of various imaging devices and conditions.

The second group of contributions is presented in Chapters 5 and 7 and relates to introducing new shape descriptors and associated methods for their numerical evaluation. In this sense, two new original descriptors of shape have been introduced, such as hexagonality and fuzzy squareness, and presented the methods of their numerical evaluation that should answer the question to which degree the shape observed

satisfies a property considered. In the thesis, we considered most of the elementary properties of the descriptors introduced and their corresponding measures from both a theoretical and empirical standpoint. All the new introduced shape descriptors are naturally defined and theoretically, i.e., mathematically, well-founded. Such a property is highly desirable from the aspect of the application since then their behaviour can be somewhat understood and anticipated in advance. This latter can also be of particular benefit in the various shape-based object analysis tasks, where, even before performing a particular analysis task, it can be predicted how the results obtained will behave, given that the shape information that has been incorporated by them. In the sequel, we will summarize the concrete contributions of the thesis, and also present several directions and potential topics of future research.

In Chapter 4, we have presented two novel methods for estimating 3D Euclidean distances in the corresponding distance map utilizing the information available in the voxel coverage image representation. In such an image representation, assigned coverage values have been observed to estimate the object position within the boundary voxels. We have shown that such an estimated sub-voxel position of the object boundary can contribute to accuracy improvement of the estimated distances if the corresponding normal direction of the object boundary is considered appropriately. Depending on the way how the information about the normal direction is involved in the distance computation, we come to two new 3D EDT estimation methods. The statistical evaluation of the proposed methods have been performed as well, comparing them both with the classical binary EDT [18], and the method relying only on the voxel coverage and object position within the boundary voxel [91]. Experimental evaluation presented indicate that significant improvement in accuracy and precision has been achieved if the proposed EDT estimates are applied. One of them, denoted as VCEDT3, exceeds the other one, denoted as VCEDT2, by reducing the errors of estimates, as well as the range of errors, while both proposed exceed binary EDT (BEDT) and the voxel coverage based EDT proposed in [91] (VCEDT1), by reduction the mean absolute error (MAE), as defined in (4.9), up to fourteen times relative to BEDT, and up to seven times with respect to VCEDT1. Besides, the complexity of the proposed EDT estimates is similar to other EDT approximations that are based on the chamfer or vector distance propagation as well. More precisely, their complexity is optimal, i.e., linear with respect to the number of image elements, providing such a reasonable approximation of the exact EDT at a reasonable (i.e., optimal) computational cost. Exploiting all the considerations presented, one of the possible open topics for future research might be designing and developing the new, or appropriate adaptation of already existing analysis tools that could have an advantage from the proposed EDT approximations. This also motivates their further evaluation in various image processing applications and tasks performed on the images obtained under different imaging conditions.

Chapter 5 introduces a new measure for evaluating how much a given shape is

hexagonal. It can be said that this is the first attempt to define such a measure, given that, according to the authors' best knowledge, it is not known yet whether there exist similar methods for measuring hexagonality of a given shape. Taking into account that hexagonality can be understood as a new shape descriptor of the object, which has a clear geometric meaning, it is natural to define hexagonality by adjusting some general procedures for defining other already known shape descriptors (see, for example, [121]). Such a defined hexagonality measure, herein denoted by $\mathcal{H}_{fit}(S)$, has most of the basic properties expected to be satisfied by all hexagonality measures. However, we have shown that for such a measure there exists a shape, with the nonzero area, whose measured hexagonality is equal to 0 (e.g., the last shape in Fig. 5.6). Such a property of $\mathcal{H}_{fit}(S)$ can be considered to be a weakness, given that, according to our initial idea, a measure of shape hexagonality should define how much a given shape differs from a hexagon.

Having in mind all these facts, we have derived in Chapter 5 a new measure of shape hexagonality, $\mathcal{H}(S)$, which has been defined in (5.21). Also, the values of $\mathcal{H}(S)$ have been shown to vary within the interval $(0,1]$, with the largest possible value of 1 being reached if and only if for a hexagon. We have shown that the new measure of hexagonality $\mathcal{H}(S)$ can also be computed using the corresponding geometric moments (5.2), which enables an easy and straightforward numerical computation of $\mathcal{H}(S)$. Besides, a new shape hexagonality is invariant concerning the geometrical transformations of rotation, translation, and scaling. All these desirable properties of $\mathcal{H}(S)$ have been also well-founded and theoretically verified. This latter is of particular interest in a number of object analysis tasks, given that in these situations the behaviour of $\mathcal{H}(S)$ can be predicted in advance to some extent. The experiments presented also provide an illustration of the behaviour of a new hexagonality measure as well as its comparison with the behaviour of $\mathcal{H}_{fit}(S)$.

A new measure of hexagonality $\mathcal{H}(S)$ also provides several useful consequences, which have been theoretically considered, and also illustrated through several experiments. First, a new hexagonality measure provides a new approach to computing the shape orientation, where the orientation is defined by direction (i.e., angle) which minimizes the integral in (5.21). Also, we have shown that a new hexagonality measure provides a new method for measuring shape elongation. The hexagonality based shape elongation is given as the ratio of the lengths of the longer and shorter semiaxis of the corresponding fitted hexagon, as given in (5.9). Such a defined elongation measure, herein denoted as $\mathcal{E}(S)$, is invariant to rotation, translation, and scaling transformations. Also, it achieves the minimum possible value of 1 if and only if the corresponding fitted hexagon is degenerative (i.e., a square). Beside all these desirable properties, an important benefit of our new approach for measuring the shape orientation and shape elongation is that they can be successfully applied to shapes for which the standard methods [152], based on the shapes moments, do not provide a clear result. This is especially noticeable for shapes with more than one axis of

symmetry, or order of rotational symmetry greater than two. We have shown that our new measures can deal with such challenges, which can be considered as their very desirable property. Moreover, this is one of the main reasons why defining a new hexagonality based method for computing the orientation and elongation of shape makes sense. Also, we have illustrated that the use of our new methods does not exclude the use of the corresponding standard methods. For example, we have shown that, for the shapes that are naturally oriented (e.g., symmetric shapes having only one axis of symmetry, elongated shapes whose orientation is determined by some axis of their elongation, etc.), our new approach provides orientations that are similar or closely aligned with the orientations obtained using the standard method. On the other hand, we have shown that for the shapes with several symmetry axes, or which are rotationally symmetric of an order greater than two, our new method is compatible with the standard moment-based method [152], given that the computed orientations are approximately matched with some of their symmetry axes. This latter is visible in the case of shape elongation since the standard method does not provide a clear distinction among the different shapes which are N-fold rotationally symmetric. This follows from the fact that for such shapes the standard method assigns elongation equals 1, whereas a new elongation measure $\mathcal{E}(S)$ assigns different measured elongations. Such a property of $\mathcal{E}(S)$ can be perceived to be an advantage over the standard elongation measure.

A new hexagonality measure $\mathcal{H}(S)$ can be efficiently and successfully applied, together with hexagonality $\mathcal{H}_{fit}(S)$ and new elongation measure $\mathcal{E}(S)$, in a wide range of image processing and computer vision tasks. To illustrate applicability and quality of the introduced shape measures in such tasks, we have considered several classification experiments on three well-known image datasets: MPEG-7 CE-1 dataset [73], Swedish Leaf dataset [151] and Galaxy Zoo database [92].The results achieved demonstrated the utility and importance of the measures introduced in all the tasks considered, even though we have used only a few simple image processing tools, a small number of shape-based measures, and very simple applied classifier. It is worth mentioning that in some experiments such as those related to MPEG-7 CE-1 and Swedish Leaf datasets, our focus has been on achieving a higher classification accuracy compared to what was achieved so far using the feature vector spaces of similar dimensionality. Consequently, we have shown that the best scores (i.e., benchmark results) were achieved if the new shape measures have been included in the set of the appropriate initially chosen shape descriptors. In these experiments, the relative improvements by including new shape measures range from approximately 3% for Swedish Leaf dataset to closely 9% what has been achieved for MPEG-7 CE-1 dataset. Based on this, we have concluded that introduced shape measures are complementary, and they can be used together to increase efficiency in these classification tasks. Regarding the third discussed image dataset, it should be emphasized that in this experiment we did not focus on the benchmark result, but we intended to further

illustrate the quality and benefits of the new shape measures. As such, this experiment can be understood as a good indicator that the applicability of our new measures in this classification task makes sense. Besides, although we have not achieved a benchmark result, the results obtained are comparable to those in the literature available from the other authors, which has been mentioned in the chapter as well. It is worth noting that this is something expected, given that there are no measures that perform better than others in all applications. Based on all the observations presented in Chapter 5, one of the interesting topics for our future research would be related to suitable generalizations of the new hexagonality measure in diverse image analysis tasks. Such generalizations would allow us to introduce several more new shape descriptors, which is of particular interest, given that, due to a permanent demand in various new image processing based tasks, it is necessary to design more and more different, but also efficient shape analysis tools.

In Chapter 6, we have derived a new estimation method of the centroid distance signature of two-dimensional shape utilizing coverage information contained in its discrete representation. Similarly to the case of three-dimensional EDTs given in Chapter 4, the intensity value assigned now to each pixel is used to estimate the boundary position of the object within the pixel. Under the assumption that the boundary of the object is locally linear and intersects the object pixel along the straight line, the midpoint of the estimated edge has been utilized to define the signature of a shape. In this chapter, we have presented an iterative procedure which improves the edge direction estimation, and consequently, the signature estimate in each iterative step. Performance evaluation has shown the advantages of the presented estimation method with respect to variance, bias, robustness to noise, as well as rotational and translational invariance, relative to both signature estimated from the sharp (i.e., binary) representation of shape, and signature computed as the average of estimated signatures over all the α-cuts of the corresponding coverage shape representation [26]. The iterative algorithm presented is simple, easy to understand and fast to compute, and, if it converges, the convergence is very fast. The experiments presented illustrate as well that the assumption regarding local linearity of the object boundary is of essential importance in designing the estimation method itself. In such cases, the algorithm presented converges, and it does it very fast. Finally, we can conclude that this chapter, similarly as Chapter 4, clearly shows that the information available in the coverage image representation can provide estimates of the shape descriptors with increased precision and accuracy. The results presented, both theoretically and empirically, encourage and motivate further research in the tasks of extending the applicability of the proposed signature estimation method to more complex shapes, as well as to the shapes obtained under the various image acquisition conditions. This is pretty important in object analysis tasks (e.g., object recognition, object classification, object retrieval, etc.) where improved performance of the proposed signature estimate can be of particular and/or essential importance.

This remains an open topic for future work.

Chapter 7 has followed the same line as Chapter 5, but now providing a new fuzzy squareness measure which defines the degree of how much a fuzzy shape considered is fuzzy squared. Given that fuzzy squareness can be recognized to be a new descriptor of the fuzzy object given, having a clear shape interpretation, one of the possibilities for its definition relates to appropriate dealing with the approach already discussed in Chapter 5, which was originally presented in [121]. We have shown that such an introduced fuzzy squareness, herein denoted as $\mathcal{FS}_{fit}(S)$, satisfies most of the elementary properties that each well-designed measure should satisfy. However, for such a defined measure there exist shapes, with the non-zero areas, whose assigned fuzzy squareness is equal to 0. For example, we have presented four such shapes in Fig. 5.6 to make this property clearer for understanding. Given that such a property of $\mathcal{FS}_{fit}(S)$ can be considered to be a disadvantage, we have derived a new measure of fuzzy squareness, $\mathcal{FS}_q(S)$, which takes the values from $(0,1]$ and achieves the maximum value 1 if and only if for a fuzzy square. We have proven that the new fuzzy squareness measure is additionally invariant to transformations of similarity. All such desirable properties of $\mathcal{FS}_q(S)$ have been theoretically proven and experimentally verified. Several illustrative experiments illustrating the behaviour of $\mathcal{FS}_q(S)$ have been included in the chapter as well.

Effectiveness and usability of the new fuzzy squareness measure $\mathcal{FS}_q(S)$ have been demonstrated through various shape analysis and object recognition tasks. For this purpose, we have limited ourselves to the experiments illustrating the benefits and quality of the new fuzzy squareness shape measures in the tasks of classifying the images from three well-known, and in the literature widely studied, image datasets. Among them there are two already used datasets in Chapter 5 such as MPEG-7 CE-1 dataset [73] and Swedish Leaf dataset [151], but also the new one Portuguese Leaf dataset [141]. The efficiencies reached have proven that the new fuzzy squareness measures can be of particular importance in these classification tasks, even if only a few features have been used with simpler applied machine learning classifier. As for the experiments performed on MPEG-7 CE-1 dataset, our goal was to provide a higher efficiency relative to scores achieved to date using the feature vectors of approximately equal dimensionality. We have shown that the highest efficiency (i.e., benchmark result) was achieved if the new squareness shape measures were used together with the initial, appropriately chosen, shape descriptors. In these experiments, the relative improvements in efficiency range approximately from 5% to 7%. Following the experiments presented, we have concluded that the new shape measures are compatible, and can be applied together to improve the efficiency in such a classification task. Regarding the second and third considered datasets, we have not aimed at a benchmark result, but to represent the capability of fuzzy squareness measures to improve the classification efficiency being performed on these datasets. As such, these experiments are presented to illustrate that our new fuzzy squareness measures

make sense in such defined classification tasks (e.g., relative improvements in accuracy are in the range from approximately 12% to 15.5% if the new fuzzy squareness measures are added into consideration). Besides, although we have not achieved a benchmark result, the results obtained are comparable to those reported in the literature to date provided by the others, and which has also been discussed in the chapter. This is something that can be expected because, as already mentioned several times, there is no shape measure superior to the others in all applications. Therefore, one direction of our further work relates to adopting the approach presented in this chapter to develop several new fuzzy-based shape measures, which can be of particular importance for shape-based analysis applications. This latter follows from the fact that, due to the increasing needs in various tasks, it is pretty important to design new simpler shape-based object analysis tools, aimed to strengthen the already existing object descriptors that have found their application in diverse image processing and computer vision tasks.

Appendix

Appendix

One of the important steps in deriving an efficient EDT algorithm in 3D with subvoxel precision in Chapter 4 was to solve the following third-degree algebraic equation

$$ax^3 + bx^2 + cx + d = 0, \tag{1}$$

over the field of complex numbers $\mathbb{C}$, where $a, b, c, d \in \mathbb{R}, a \neq 0$ and $x \in \mathbb{C}$. For such a task, we have utilized Cardano formula, named after Girolamo Cardano (1501-1576), which was first published in his book *Ars Magna* in 1545. Because of its importance, we herein present its derivation to give the reader the ability to fully understand it.

At first, by multiplying the initial equation (1) by a^{-1}, the following *normalized cubic equation* is obtained

$$x^3 + Ax^2 + Bx + C = 0, \tag{2}$$

where $A = \frac{b}{a}$, $B = \frac{c}{a}$ and $C = \frac{d}{a}$. Further, if the linear substitution $y = x + \frac{A}{3}$ is applied to (2), it is reduced to

$$\left(y - \frac{A}{3}\right)^3 + A\left(y - \frac{A}{3}\right)^2 + B\left(y - \frac{A}{3}\right) + C = 0$$

$$\Leftrightarrow y^3 + \left(B - \frac{A^2}{3}\right)y + \left(\frac{2A^3}{27} - \frac{AB}{3} + C\right) = 0, \tag{3}$$

which can be written in *canonical form*

$$y^3 + py + q = 0, \tag{4}$$

where $p = B - \frac{A^2}{3}$, $q = \frac{2A^3}{27} - \frac{AB}{3} + C$. After introducing the substitution

$$y = u + v, \tag{5}$$

equation (4) becomes

$$\begin{aligned}(u+v)^3 + p(u+v) + q = 0 \\ \Leftrightarrow u^3 + v^3 + (3uv+p)(u+v) + q = 0,\end{aligned} \tag{6}$$

where u and v are now the new variables. Given that each variable can be written as a sum of two variables in infinitely many ways, we can introduce additional condition that the new variables should satisfy. For example, if we involve the condition $3uv + p = 0$, then the equation (6) becomes

$$u^3 + v^3 = -q, \tag{7}$$

providing thus the following system of equations:

$$u^3 + v^3 = -q, \qquad uv = -\frac{p}{3}, \tag{8}$$

i.e.,

$$u^3 + v^3 = -q, \qquad u^3 v^3 = -\left(\frac{p}{3}\right)^3. \tag{9}$$

However, taking into account that the latter relations represent the sum and product of the monomials u^3 and v^3 then, according to Vieta's formulas, there exists a quadratic equation whose roots are these monomials, i.e.,

$$z^2 - (u^3 + v^3)z + u^3 v^3 = 0, \tag{10}$$

or, equivalently

$$z^2 + qz - \left(\frac{p}{3}\right)^3 = 0, \tag{11}$$

whereas its corresponding roots are given as follows:

$$\begin{aligned} z_1 &= u^3 = -\frac{q}{2} + \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}, \\ z_2 &= v^3 = -\frac{q}{2} - \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}. \end{aligned} \tag{12}$$

Following this, the roots of the cubic equation (4) are then determined as

$$y = \sqrt[3]{-\frac{q}{2} + \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}} + \sqrt[3]{-\frac{q}{2} - \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}}, \tag{13}$$

which is also known as *Cardano formula*, whereas the expression under the square root $D = \left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3$ is called the *discriminant* of the cubic equation (4). Denoting with $R = -\frac{q}{2}$ and $Q = \frac{p}{3}$, the Cardano formula (13) can be then written in a more compact form as follows:

$$y = \sqrt[3]{R + \sqrt{R^2 + Q^3}} + \sqrt[3]{R - \sqrt{R^2 + Q^3}}. \tag{14}$$

Given that each third degree polynomial over the field $\mathbb{C}$ has at least one real root, and denoting with $u_1 = \sqrt[3]{R + \sqrt{R^2 + Q^3}}$ and $v_1 = \sqrt[3]{R - \sqrt{R^2 + Q^3}}$ a pair of the real roots of equations $u = \sqrt[3]{u^3}$ and $v = \sqrt[3]{v^3}$, the sets of all the complex roots for u and v are given as follows:

$$\begin{aligned} u &\in \{u_1,\ u_1 e^{\frac{2\pi}{3}i},\ u_1 e^{-\frac{2\pi}{3}i}\}, \\ v &\in \{v_1,\ v_1 e^{\frac{2\pi}{3}i},\ v_1 e^{-\frac{2\pi}{3}i}\}. \end{aligned} \tag{15}$$

This further implies that equation (4) has at most nine potential roots, given in the form $y = u + v$. However, given that its roots must also satisfy the condition $uv = -Q$, the roots of (4) are given as follows:

$$y = \begin{cases} u_1 + v_1 & \\ u_2 + v_3 & = -\frac{u_1+v_1}{2} + \frac{i\sqrt{3}}{2}(u_1 - v_1) \\ u_3 + v_2 & = -\frac{u_1+v_1}{2} - \frac{i\sqrt{3}}{2}(u_1 - v_1), \end{cases} \tag{16}$$

whereas the roots of (1) are then expressed in the following way:

$$x = \begin{cases} u_1 + v_1 - \frac{b}{3a} \\ -\frac{u_1+v_1}{2} - \frac{b}{3a} + \frac{i\sqrt{3}}{2}(u_1 - v_1) \\ -\frac{u_1+v_1}{2} - \frac{b}{3a} - \frac{i\sqrt{3}}{2}(u_1 - v_1), \end{cases} \tag{17}$$

where

$$u_1 = \sqrt[3]{R + \sqrt{Q^3 + R^2}}, \qquad v_1 = \sqrt[3]{R - \sqrt{Q^3 + R^2}},$$

and

$$Q = \frac{3ac - b^2}{9a^2}, \qquad R = \frac{9abc - 27a^2d - 2b^3}{54a^3}.$$

As it can be noticed, this latter represents the form of the solution of the initial cubic equation (1) used in the Algorithm 1 in Chapter 4. Besides, it is worth noting that,

similarly as for the quadratic equation, the nature of the roots of cubic equation (1) can be determined in advance, without their solving, observing only the sign of the associated discriminant $D = Q^3 + R^2$. Taking into account these considerations, the three different cases are possible:

$$\begin{aligned} &1) \quad \text{for } D > 0, \text{ one root is real and other two are complex conjugates,} \\ &2) \quad \text{for } D = 0, \text{ all the roots are real and at least two are equal,} \\ &3) \quad \text{for } D < 0, \text{ all the roots are real and unequal.} \end{aligned} \tag{18}$$

Bibliography

[1] Car logos. https://www.carlogos.org/.

[2] Car logos. https://www.carlogos.org/tire-brands/.

[3] N. Alajlan, I. E. Rubé, M. S. Kamel, and G. H. Freeman. Shape retrieval using triangle-area representation and dynamic space warping. *Pattern Recognition*, 40(7):1911–1920, 2007.

[4] N. Alajlan, M. S. Kamel, and G. H. Freeman. Geometry-based image retrieval in binary image databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1003–1013, 2008.

[5] S. Angel, J. Parent, and D. Civco. Ten compactness properties of circles: measuring shape in geography. *The Canadian Geographer*, 54(4):441–461, 2010.

[6] C. Au. A simple algorithm for medial axis transform computation. *Engineering with Computers*, 29(2):139–149, 2013.

[7] A. R. Backes and O. M. Bruno. Shape classification using complex network and multi-scale fractal dimension. *Pattern Recognition Letters*, 31(1):44–51, 2010.

[8] A. R. Backes and O. M. Bruno. Polygonal approximation of digital planar curves through vertex betweenness. *Information Sciences*, 222:795–804, 2013.

[9] A. R. Backes, D. Casanova, and O. M. Bruno. A complex network-based approach for boundary shape analysis. *Pattern Recognition*, 42(1):54–67, 2009.

[10] X. Bai, W. Liu, and Z. Tu. Integrating contour and skeleton for shape classification. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 360–367. IEEE, 2009.

[11] P. Barré, B. C. Stöver, kai F. Müller, and V. Steinhage. Leafnet: A computer vision system for automatic plant species identification. *Ecological Informatics*, 40:50–56, 2017.

[12] I. Bartolini, P. Ciaccia, and M. Patella. WARP: accurate retrieval of shapes using phase of Fourier descriptors and time warping distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1):142–147, 2005.

[13] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.

[14] S. Berkaya, H. Gunduz, O. Ozsen, C. Akinlar, and S. Gunal. On circular traffic sign detection and recognition. *Expert Systems with Applications*, 48:67–75, 2016.

[15] I. Bloch. Fuzzy sets for image processing and understanding. *Fuzzy Sets and Systems*, 281:280–291, 2015.

[16] H. Blum. A transformation for extracting new descriptors of shape. In W. W. Dunn, editor, *Models for the Perception of Speech and Visual Form*, pages 362–381. MIT Press, 1967.

[17] A. Bogomolny. On the perimeter and area of fuzzy sets. *Fuzzy Sets and Systems*, 23(2):257–269, 1987.

[18] G. Borgefors. Distance transformations in arbitrary dimensions. *Computer Vision, Graphics and Image Processing*, 27(3):321–345, 1984.

[19] G. Borgefors. Distance transformations in digital images. *Computer Vision, Graphics and Image Processing*, 34(3):344–371, 1986.

[20] G. Borgefors. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(6):849–865, 1988.

[21] G. Borgefors. Applications using distance transforms. In *Aspects of Visual Form Processing*, pages 83–108. World Scientific Singapore, 1994.

[22] G. Borgefors. On digital distance transforms in three dimensions. *Computer Vision and Image Understanding*, 64(3):368–376, 1996.

[23] G. Borgefors and I. Nyström. Efficient shape representation by minimizing the set of centres of maximal discs/spheres. *Pattern Recognition Letters*, 18 (5):465–471, 1997.

[24] J. J. Buckley and E. Eslami. Fuzzy plane geometry I: points and lines. *Fuzzy Sets and Systems*, 86(2):179–187, 1997.

[25] M. A. Butt and P. Maragos. Optimum design of chamfer distance transforms. *IEEE Transactions on Image Processing*, 7(10):1477–1484, 1998.

[26] J. Chanussot, I. Nyström, and N. Sladoje. Shape signatures of fuzzy star-shaped sets based on distance from the centroid. *Pattern Recognition Letters*, 26(6):735–746, 2005.

[27] S. Chattopadhyay. A neuro-fuzzy approach for the diagnosis of depression. *Applied Computing and Informatics*, 13(1):10–18, 2017.

[28] S. Chetia and S. R. Nirmala. Polynomial modeling of retinal vessels for tortuosity measurement. *Biocybernetics and Biomedical Engineering*, 39(2):512–525, 2019.

[29] Y. T. Chin, H. Wang, L. P. Tay, H. Wang, and W. Y. C. Soh. Vision guided AGV using distance transform. In *Proceedings 32nd International Symposium on Robotics - ISR 2001, in conjunction with IMS 2001 1st Intelligent Microsystem Symposium*, volume 2, pages 1416–1421, 2001.

[30] M. Clément, C. Kurtz, and L. Wendling. Bags of spatial relations and shapes features for structural object description. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 1994–1999. IEEE, 2016.

[31] M. Clément, C. Kurtz, and L. Wendling. Learning spatial relations and shapes for structural object description and scene recognition. *Pattern Recognition*, 84:197–210, 2018.

[32] D. Coeurjolly and A. Montanvert. Optimal separable algorithms to compute the reverse Euclidean distance transformation and discrete medial axis in arbitrary dimension. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):437–448, 2007.

[33] M. Couprie, A. V. Saúde, and G. Bertrand. Euclidean homotopic skeleton based on critical kernels. In *2006 19th Brazilian Symposium on Computer Graphics and Image Processing*, pages 307–314. IEEE, 2006.

[34] R. A. C. Croft, P. E. Freeman, T. S. Schuster, and C. M. Schafer. Prediction of galaxy ellipticities and reduction of shape noise in cosmic shear measurements. *Monthly Notices of the Royal Astronomical Society*, 469(4):4422–4427, 2017.

[35] O. Cuisenaire. Locally adaptable mathematical morphology using distance transformations. *Pattern Recognition*, 39(3):405–416, 2006.

[36] O. Cuisenaire and B. Macq. Fast Euclidean distance transformation by propagation using multiple neighborhoods. *Computer Vision and Image Understanding*, 76(2):163–172, 1999.

[37] R. da Silva Torres, A. X. Falcão, and L. da Fontoura Costa. A graph-based approach for multiscale shape analysis. *Pattern Recognition*, 37(6):1163–1174, 2004.

[38] P.-E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14(3):227–248, 1980.

[39] P.-E. Danielsson and S. Tanimoto. Time complexity for serial and parallel propagation in images. In *Proceedings SPIE*, volume 435 of *Architecture and Algorithms for Digital Image Processing*, pages 60–67, 1984.

[40] A. B. de Oliveira, P. R. da Silva, and D. A. C. Barone. A novel 2D shape signature method based on complex network spectrum. *Pattern Recognition Letters*, 63:43–49, 2015.

[41] M. M. de Souza, F. N. de Medeiros, G. L. B. Ramalho, I. C. de Paula Jr., and I. N. Oliveira. Evolutionary optimization of a multiscale descriptor for leaf shape analysis. *Expert Systems With Applications*, 63:375–385, 2016.

[42] A.-W. Deng and C.-Y. Gwo. Fast and stable algorithms for high-order Pseudo Zernike moments and image reconstruction. *Applied Mathematics and Computation*, 334:239–253, 2018.

[43] A.-W. Deng, C.-H. Wei, and C.-Y. Gwo. Stable, fast computation of high-order Zernike moments using a recursive method. *Pattern Recognition*, 56: 16–25, 2016.

[44] R. S. Dewi, W. Bijker, and A. Stein. Comparing fuzzy sets and random sets to model the uncertainty of fuzzy shorelines. *Remote Sensing*, 9(9), 2017.

[45] P. Diamond. A note on fuzzy starshaped fuzzy sets. *Fuzzy Sets and Systems*, 37(2):193–199, 1990.

[46] C. Direkŏglu and M. S. Nixon. Shape classification via image-based multiscale description. *Pattern Recognition*, 44(9):2134–2146, 2011.

[47] I. Djurić, V. Ilić, and N. Ralević. Shape descriptors applied to the analysis of different types of the engineering elements. *Journal of Production Engineering*, 22(1):29–34, 2019.

[48] S. Dražić, N. Ralević, and J. Žunić. Shape elongation from optimal encasing rectangles. *Computers and Mathematics with Applications*, 60(7):2035–2042, 2010.

[49] R. E. González, R. Muñoz, and C. A. Hernández. Galaxy detection and identification using deep learning and data augmentation. *Astronomy and Computing*, 25:103–109, 2018.

[50] A. El-ghazal, O. A. Basir, and S. Belkasim. A new shape signature for Fourier descriptors. In *2007 IEEE International Conference on Image Processing*, volume 1, pages I – 161–I – 164. IEEE, 2007.

[51] A. El-ghazal, O. A. Basir, and S. Belkasim. Farthest point distance: A new shape signature for Fourier descriptors. *Signal Processing: Image Communication*, 24(7):572–586, 2009.

[52] A. Ellahyani, M. E. Ansari, and I. E. jaafari. Traffic sign detection and recognition based on random forests. *Applied Soft Computing*, 46:805–815, 2016.

[53] R. Fabbri, L. da Fontoura Costa, J. C. Torelli, and O. M. Bruno. 2D Euclidean distance transform algorithms: A comparative survey. *ACM Computing Surveys*, 40(1):2:1–2:44, 2008.

[54] P. F. Felzenszwalb and J. D. Schwartz. Hierarchical matching of deformable shapes. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[55] P. P. R. Filho, E. de S. Rebouças, L. B. Marinho, R. M. Sarmento, J. M. R. S. Tavares, and V. H. C. de Albuquerque. Analysis of human tissue densities: A new approach to extract features from medical images. *Pattern Recognition Letters*, 94:211–218, 2017.

[56] P. P. R. Filho, F. D. L. Moreira, L. B. Marinho, and J. S. Almeida. Novel and powerful mosaic constructor for territorial analysis using mobile robots via binary robust invariant scalable keypoints. *Measurement*, 114:448–470, 2018.

[57] J. Flusser and T. Suk. Pattern recognition by affine moment invariants. *Pattern Recognition*, 26(1):167–174, 1993.

[58] B. Garg, S. Aggarwal, and J. Sokhal. Crop yield forecasting using fuzzy logic and regression model. *Computers & Electrical Engineering*, 67:383–403, 2018.

[59] S. Garrido, L. Moreno, D. Blanco, and P. P. Jurewicz. Path planning for mobile robot navigation using Voronoi diagram and fast marching. *International Journal of Intelligent Systems and Applications in Robotics*, 2(1):42–64, 2011.

[60] M. Gong, H. Li, and W. Cao. Moment invariants to affine transformation of colours. *Pattern Recognition Letters*, 34(11):1240–1251, 2013.

[61] G. J. Grevera. Distance transform algorithms and their implementation and evaluation. In J. S. Suri and A. A. Farag, editors, *Deformable Models*, Topics in Biomedical Engineering. International Book Series, pages 33–60. Springer, New York, NY, 2007.

[62] A. Gudigar, S. Chokkadi, U.Raghavendra, and U. R. Acharya. Local texture patterns for traffic sign recognition using higher order spectra. *Pattern Recognition Letters*, 94:202–210, 2017.

[63] S. Gustavson and R. Strand. Anti-aliased Euclidean distance transform. *Pattern Recognition Letters*, 32(2):252–257, 2011.

[64] G. H.B.Miranda, J. Machicao, and O. M.Bruno. An optimized shape descriptor based on structural properties of networks. *Digital Signal Processing*, 82:216–229, 2018.

[65] B. Hoyle. Measuring photometric redshifts using galaxy images and deep neural networks. *Astronomy and Computing*, 16:34–40, 2016.

[66] M.-K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187, 1962.

[67] B. Huang, J. Wu, D. Zhang, and N. Li. Tongue shape classification by geometric features. *Information Sciences*, 180(2):312–324, 2010.

[68] V. Ilić. Distance transforms and their applications in robot navigation. In *The Third Conference on Mathematics in Engineering: Theory and Application*, pages 92–97, 2018.

[69] V. Ilić and N. M. Ralević. Hexagonality as a new shape based descriptor of object. *Submitted*, .

[70] V. Ilić and N. M. Ralević. Fuzzy-squareness: a new approach for measuring a shape. *Submitted*, .

[71] V. Ilić, J. Lindblad, and N. Sladoje. Precise Euclidean distance transforms in 3D from voxel coverage representation. *Pattern Recognition Letters*, 65: 184–191, 2015.

[72] V. Ilić, J. Lindblad, and N. Sladoje. Signature of a shape based on its pixel coverage representation. In N. Normand, J. Guédon, and F. Autrusseau, editors, *Discrete Geometry for Computer Imagery, DGCI 2016*, volume 9647 of *Lecture Notes in Computer Science*, pages 181–193. Springer, Cham, 2016.

[73] L. Jan Latecki, R. Lakämper, and T. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2000 (Cat. No.PR00662)*, volume 1, pages 424–429. IEEE, 2000.

[74] M. W. Jones, J. A. Bærentzen, and M. Šrámek. 3D distance fields: A survey of techniques and applications. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):581–599, 2006.

[75] J. K. Udupa and G. J. Grevera. Go digital, go fuzzy. In S. Pal, S. Bandyopadhyay, and S. Biswas, editors, *Pattern Recognition and Machine Intelligence, PReMI 2005*, volume 3776 of *Lecture Notes in Computer Science*, pages 137–146. Springer, Berlin, Heidelberg, 2005.

[76] A. A. Kalitzeos, G. Y. H. Lip, and R. Heitmar. Retinal vessel tortuosity measures and their applications. *Experimental Eye Research*, 106:40–46, 2013.

[77] C. Kalyoncu and Ö. Toygar. Geometric leaf classification. *Computer Vision and Image Understanding*, 133:102–109, 2015.

[78] V. V. Kindratenko. On using functions to describe the shape. *Journal of Mathematical Imaging and Vision*, 18:225–245, 2003.

[79] R. Klette and J. D. Žunić. On discrete moments of unbounded order. In A. Kuba, L. G. Nyúl, and K. Palágyi, editors, *Discrete Geometry for Computer Imagery, DGCI 2006*, volume 4245 of *Lecture Notes in Computer Science*, pages 367–378. Springer, Berlin, Heidelberg, 2006.

[80] D. Kozinska, O. J. Tretiak, J. Nissanov, and C. Ozturk. Multidimensional alingment using the Euclidean distance transform. *Graphical Models Image Processing*, 59(6):373–387, 1997.

[81] I. Kunttu and L. Lepistö. Shape-based retrieval of industrial surface defects using angular radius Fourier descriptor. *IET Image Processing*, 1(2):231–236, 2007.

[82] I. Kunttu, L. Lepistö, J. Rauhamaa, and A. Visa. Multiscale Fourier descriptors for defect image retrieval. *Pattern Recognition Letters*, 27(2):123–132, 2006.

[83] L. Kurnianggoro, W. Wahyono, and K.-H. Jo. A survey of 2D shape representation: Methods, evaluations, and future research directions. *Neurocomputing*, 300:1–16, 2018.

[84] S. H. Lee, C. S. Chan, S. Mayo, and P. Remagnino. How deep learning extracts and learns leaf features for plant classification. *Pattern Recognition*, 71:1–13, 2017.

[85] F. Leymarie and M. D. Levine. Fast raster scan distance propagation on the discrete rectangular lattice. *CVGIP: Image Understanding*, 55(1):84–94, 1992.

[86] R. Li, X. Shi, Y. Sheng, and G. Zhang. A new area-based convexity measure with distance weighted area integration for planar shapes. *Computer Aided Geometric Design*, 71:176–189, 2019.

[87] S. Li, M.-C. Lee, and D. A. Adjeroh. Effective invariant features for shape-based image retrieval. *Journal of the American Society for Information Science and Technology*, 56(7):729–740, 2005.

[88] W. Li, M. F. Goodchild, and R. L. Church. An efficient measure of compactness for two-dimensional shapes and its application in regionalization problems. *International Journal of Geographical Information Science*, 27(6): 1227–1250, 2013.

[89] J. Lindblad and N. Sladoje. Coverage segmentation based on linear unmixing and minimization of perimeter and boundary thickness. *Pattern Recognition Letters*, 33(6):728–738, 2012.

[90] H. Ling and D. W. Jacobs. Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2):286–299, 2007.

[91] E. Linnér and R. Strand. Anti-aliased Euclidean distance transform on 3D sampling lattices. In E. Barcucci, A. Frosini, and S. Rinaldi, editors, *Discrete Geometry for Computer Imagery, DGCI 2014*, volume 8668 of *Lecture Notes in Computer Science*, pages 88–98. Springer, Cham, 2014.

[92] C. J. Lintott, K. Schawinski, P. F. Scott, K. R. Land, S. Bamford, D. I. Thomas, M. J. Raddick, R. C. Nichol, A. S. Szalay, D. Andreescu, P. Murray, and J. B. van den Berg. Galaxy Zoo: Morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey. *Monthly Notices of the Royal Astronomical Society*, 389(3):1179–1189, 2008.

[93] Z. Liu and P. J. Keller. Emerging imaging and genomic tools for developmental systems biology. *Developmental Cell*, 36(6):597–610, 2016.

[94] S. Lončarić. A survey of shape analysis techniques. *Pattern Recognition*, 31 (8):983–1001, 1998.

[95] J. Lu, W. S. Lee, H. Gan, and X. Hu. Immature citrus fruit detection based on local binary pattern feature and hierarchical contour analysis. *Biosystems Engineering*, 171:78–90, 2018.

[96] A. M. Maceachren. Compactness of geographic shape: Comparison and evaluation of measures. *Geografiska Annaler: Series B, Human Geography*, 67 (1):53–67, 1985.

[97] F. Malmberg, J. Lindblad, N. Sladoje, and I. Nyström. A graph-based framework for sub-pixel image segmentation. *Theoretical Computer Science*, 412 (15):1338–1349, 2011.

[98] L. B. Marinho, P. P. R. Filho, J. S. Almeida, J. W. M. de Souza, A. H. de Souza Júnior, and V. H. C. de Albuquerque. A novel mobile robot localization approach based on classification with rejection option using computer vision. *Computers and Electrical Engineering*, 68:26–43, 2018.

[99] C. R. Maurer, R. Qi, and V. Raghavan. A linear time algorithm for computing exact Euclidean distance transform of binary images in arbitrary dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):265–270, 2003.

[100] N. Minc and M. Piel. Predicting division plane position and orientation. *Trends in Cell Biology*, 22(4):193–200, 2012.

[101] F. Mokhtarian and M. Bober. *Curvature scale space representation: theory, applications, and MPEG-7 standardization*, volume 25. Springer Science & Business Media, 2013.

[102] M. Momin, M.T.Rahman, M.S.Sultana, C.Igathinathane, A. Ziauddin, and T.E.Grift. Geometry-based mass grading of mango fruits using image processing. *Information Processing in Agriculture*, 4(2):150–160, 2017.

[103] U. Montanari. A method for obtaining skeletons using a quasi-Euclidean distance. *Journal of the Association for Computing Machinery*, 15(4):600–624, 1968.

[104] S. Mouine, I. Yahiaoui, and A. Verroust-Blondet. A shape-based approach for leaf classification using multiscale triangular representation. In *ICMR '13 - Proceedings of the 3rd ACM International Conference on Multimedia Retrieval*, pages 127–134. ACM, 2013.

[105] J. C. Mullikin. The vector distance transform in two and three dimensions. *CVGIP: Graphical Models and Image Processing*, 54(6):526–535, 1992.

[106] R. B. Oliveira, J. P. Papa, A. S. Pereira, and J. M. R. S. Tavares. Computational methods for pigmented skin lesion classification in images: Review and future trends. *Neural Computing and Applications*, 29(3):613–636, 2018.

[107] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[108] D. W. Paglieroni. Distance transforms: Properties and machine vision applications. *CVGIP: Graphical Models and Image Processing*, 54(1):56–74, 1992.

[109] S.-H. Peng, D.-H. Kim, S.-L. Lee, and C.-W. Chung. A visual shape descriptor using sectors and shape context of contour lines. *Information Sciences*, 180 (16):2925–2939, 2010.

[110] G. Prabakaran, D. Vaithiyanathan, and M. Ganesan. Fuzzy decision support system for improving the crop productivity and efficient use of fertilizers. *Computers and Electronics in Agriculture*, 150:88–97, 2018.

[111] I. Ragnemalm. Neighborhoods for distance transformations using ordered propagation. *CVGIP: Image Understanding*, 56(3):399–409, 1992.

[112] I. Ragnemalm. The Euclidean distance transform in arbitrary dimensions. *Pattern Recognition Letters*, 14(11):883–888, 1993.

[113] E. Rahtu, M. Salo, and J. Heikkilä. A new convexity measure based on a probabilistic interpretation of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1501–1512, 2006.

[114] L. Ramos, J. Novo, J. Rouco, S. Romeo, M. D. Álvarez, and M. Ortega. Multi-expert analysis and validation of objective vascular tortuosity measurements. *Procedia Computer Science*, 126:482–489, 2018.

[115] M. B. H. Rhouma, J. Žunić, and M. C. Younis. Moment invariants for multi-component shapes with applications to leaf classification. *Computers and Electronics in Agriculture*, 142(Part A):326–337, 2017.

[116] L. Ribas, M. B. Neiva, and O. M. Bruno. Distace transform network for shape analysis. *Information Sciences*, 470:28–42, 2019.

[117] A. Rosenfeld and J. L. Pfaltz. Sequential operations in digital picture processing. *Journal of the Association for Computing Machinery*, 13(4):471–494, 1966.

[118] A. Rosenfeld and J. L. Pfaltz. Distance functions on digital pictures. *Pattern Recognition*, 1(1):33–61, 1968.

[119] P. L. Rosin. Measuring rectangularity. *Machine Vision and Applications*, 11 (4):191–196, 1999.

[120] P. L. Rosin. Measuring shape: Ellipticity, rectangularity, and triangularity. *Machine Vision and Applications*, 14(3):172–184, 2003.

[121] P. L. Rosin and J. Žunić. 2D shape measures for computer vision. In A. Nayak and I. Stojmenović, editors, *Handbook of Applied Algorithms: Solving Scientific, Engineering and Practical Problems*, chapter 12, pages 347–372. John Wiley & Sons, Inc., 2008.

[122] P. L. Rosin and J. Žunić. Measuring squareness and orientation of shapes. *Journal of Mathematical Imaging and Vision*, 39(1):13–27, 2011.

[123] I. E. Rubé, N. Alajlan, M. S. Kamel, M. Ahmed, and G. H. Freeman. MTAR: A robust 2D shape representation. *International Journal of Image and Graphics*, 6(3):421–444, 2006.

[124] K. S.-Y. and Y. Yang. A self-navigating robot using fuzzy petri nets. *Robotics and Autonomous Systems*, 101:153–165, 2018.

[125] J. J. d. M. Sá Junior and A. R. Backes. Shape classification using line segment statistics. *Information Sciences*, 305:349–356, 2015.

[126] S. Salti, A. Petrelli, F. Tombari, N. Fioraio, and L. di Stefano. Traffic sign detection via interest region extraction. *Pattern Recognition*, 48(4):1039–1049, 2015.

[127] R. Santiago-Montero and E. Bribiesca. State of the art of compactness and circularity measures. *International Mathematical Forum*, 4(27):1305–1335, 2009.

[128] R. Santiago-Montero, M. A. López-Morales, and J. H. Sossa. Digital shape compactness measure by means of perimeter ratios. *Electronics Letters*, 50(3): 171–173, 2014.

[129] R. Satherley and M. W. Jones. Vector-city vector distance transform. *Computer Vision and Image Understanding*, 82(3):238–254, 2001.

[130] A. V. Saúde, M. Couprie, and R. de Alencar Lotufo. Exact Euclidean medial axis in higher resolution. In A. Kuba, L. G. Nyúl, and K. Palágyi, editors, *Discrete Geometry for Computer Imagery, DGCI 2006*, volume 4245 of *Lecture Notes in Computer Science*, pages 605–616. Springer, Berlin, Heidelberg, 2006.

[131] T. B. Sebastian, P. K. Klein, and B. B. Kimia. On aligning curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):116–125, 2003.

[132] A. Sedaghat and H. Ebadi. Remote sensing image matching based on adaptive binning sift descriptor. *IEEE Transactions on Geoscience and Remote Sensing*, 53(10):5283–5293, 2015.

[133] A. Sedaghat and N. Mohammadi. Uniform competency-based local feature extraction for remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135:142–157, 2018.

[134] A. Sedaghat and N. Mohammadi. Illumination-robust remote sensing image matching based on oriented self-similarity. *ISPRS Journal of Photogrammetry and Remote Sensing*, 153:21–35, 2019.

[135] L. Shamir. Automatic morphological classification of galaxy images. *Monthly Notices of the Royal Astronomical Society*, 399(3):1367–1372, 2009.

[136] L. Shamir. Ganalyzer: A tool for automatic galaxy image analysis. *The Astrophysical Journal*, 736(2):141–148, 2011.

[137] B. H. Shekar and B. Pilar. Shape representation and classification through pattern spectrum and local binary pattern – a decision level fusion approach. In *2014 Fifth International Conference on Signal and Image Processing*, pages 218–224. IEEE, 2014.

[138] F. Y. Shih and Y.-T. Wu. Three-dimensional Euclidean distance transformation and its application to shortest path planning. *Pattern Recognition*, 37(1):79–92, 2004.

[139] X. Shu and X.-J. Wu. A novel contour descriptor for 2D shape matching and its application to image retrieval. *Image and Vision Computing*, 29(4):286–294, 2011.

[140] X. Shu, L. Pan, and X.-J. Wu. Multi-scale contour flexibility shape signature for Fourier descriptor. *Journal of Visual Communication and Image Representation*, 26:161–167, 2015.

[141] P. F. B. Silva, A. R. S. Marçal, and R. M. A. da Silva. Evalation of features for leaf discrimination. In M. Kamel and A. Campilho, editors, *Image Analysis and Recognition, ICIAR 2013*, volume 7950 of *Lecture Notes in Computer Science*, pages 197–204. Springer, Berlin, Heidelberg, 2013.

[142] C. Singh and R. Upneja. Fast and accurate method for high order Zernike moments computation. *Applied Mathematics and Computation*, 218:7759–7773, 2012.

[143] C. Singh, E. Walia, and R. Upneja. Accurate calculation of Zernike moments. *Information Sciences*, 233:255–275, 2013.

[144] C. Singh, A. Aggarwal, and S. K. Ranade. A new convolution model for the fast computation of Zernike moments. *AEU - International Journal of Electronics and Communications*, 72:104–113, 2017.

[145] V. Singh and A. Misra. Detection of plant leaf diseases using image segmentation and soft computing techniques. *Information Processing in Agriculture*, 4(1):41–49, 2017.

[146] N. Sladoje and J. Lindblad. Representation and reconstruction of fuzzy disks by moments. *Fuzzy Sets and Systems*, 158(5):517–534, 2007.

[147] N. Sladoje and J. Lindblad. High-precision boundary length estimation by utilizing gray-level information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):357–363, 2009.

[148] N. Sladoje and J. Lindblad. Pixel coverage segmentation for improved feature estimation. In P. Foggia, C. Sansone, and M. Vento, editors, *Image Analysis and Processing – ICIAP 2009*, volume 5716 of *Lecture Notes in Computer Science*, pages 929–938. Springer, Berlin, Heidelberg, 2009.

[149] N. Sladoje and J. Lindblad. The coverage model and its use in image processing. In M. Mihaljević, editor, *Selected Topics on Image Processing and Cryptology: Zbornik radova (Collection of Papers)*, volume 15, pages 39–117. Mathematical Institute of the Serbian Academy of Sciences and Arts, Belgrade, Serbia, 2012.

[150] N. Sladoje, I. Nyström, and P. K. Saha. Measurements of digitized objects with fuzzy borders in 2D and 3D. *Image and Vision Computing*, 23(2):123–132, 2005.

[151] O. Söderkvist. Computer vision classification of leaves from swedish trees. Master's thesis, Linköping University, Department of Electrical Engineering, Computer Vision, 2001.

[152] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Cengage Learning, fourth edition, 2014.

[153] T. Suk and J. Flusser. Affine moment invariants generated by graph method. *Pattern Recognition*, 44(9):2047–2056, 2011.

[154] S. Svensson and G. Borgefors. Digital distance transforms in 3D images using information from neighbourhoods up to $5\times5\times5$. *Computer Vision and Image Understanding*, 88(1):24–53, 2002.

[155] S. Svensson, G. Borgefors, and I. Nyström. On reversible skeletonization using anchor-points from distance transforms. *Journal of Visual Communication and Image Representation*, 10(4):379–397, 1999.

[156] Z. Tu and A. L. Yuille. Shape matching and recognition – using generative models and informative features. In T. Pajdla and J. Matas, editors, *Computer Vision - ECCV 2004*, volume 3023 of *Lecture Notes in Computer Science*, pages 195–209. Springer, Berlin, Heidelberg, 2004.

[157] P. van Otterloo. *A Contour-Oriented Approach to Shape Analysis*. PhD thesis, Electrical Engineering, Mathematics and Computer Science, 1988.

[158] L. Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, 1991.

[159] J. Žunić and P. L. Rosin. A new convexity measure for polygons. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):923–934, 2004.

[160] J. Žunić and D. Žunić. Shape interpretation of second-order moment invariants. *Journal of Mathematical Imaging and Vision*, 56(1):125–136, 2016.

[161] J. Žunić, K. Hirota, and P. L. Rosin. A Hu invariant as a shape circularity measure. *Pattern Recognition*, 43(1):47–57, 2010.

[162] J. Žunić, P. L. Rosin, and V. Ilić. Disconnectedness: A new moment invariant for multi-component shapes. *Pattern Recognition*, 78:91–102, 2018.

[163] P. Wan, A. Toudeshki, H. Tan, and R. Ehsani. A methodology for fresh tomato maturity detection using computer vision. *Computers and Electronics in Agriculture*, 146:43–50, 2018.

[164] B. Wang. Shape retrieval using combined Fourier features. *Optics Comunications*, 284(14):3504–3508, 2011.

[165] C. Wang, W. Xingyuan, L. Yongwei, X. Zhiqiu, and C. Zhang. Quaternion polar harmonic Fourier moments for color images. *Information Sciences*, 450: 141–156, 2018.

[166] J. Wang, X. Bai, X. You, W. Liu, and L. J. Latecki. Shape matching and classification using height functions. *Pattern Recognition Letters*, 33(2):134–143, 2012.

[167] X. Wang, B. Feng, X. Bai, W. Liu, and L. J. Latecki. Bag of contour fragments for robust shape classification. *Pattern Recognition*, 47(6):2116–2125, 2014.

[168] E. A. Wentz. A shape definition for geographic applications based on edge, elongation, and perforation. *Geographical Analysis*, 32(2):95–112, 2010.

[169] W.-T. Wong, F. Y. Shih, and J. Liu. Shape-based image retrieval using support vector machines, Fourier decriptors and self-organizing maps. *Information Sciences*, 177(8):1878–1891, 2007.

[170] W.-Y. Wu and M.-J. J. Wang. Detecting the dominant points by the curvature-based polygonal approximation. *CVGIP: Graphical Model and Image Processing*, 55(2):79–88, 1993.

[171] J. Xie, P.-A. Heng, and M. Shaha. Shape matching and modeling using skeletal context. *Pattern Recognition*, 41(5):1756–1767, 2008.

[172] D. Xu and H. Li. Geometric moment invariants. *Pattern Recognition*, 41(1): 240–249, 2008.

[173] R. B. Yadav, N. K. Nishchal, A. K. Gupta, and V. K. Rastogi. Retrieval and classification of shape-based objects using Fourier, generic Fourier, and wavelet-Fourier descriptors technique: A comparative study. *Optics and Lasers in Engineering*, 45(6):695–708, 2007.

[174] H. Yamada. Complete Euclidean distance transformation by parallel operation. volume 1, pages 69–71. IEEE, 1984.

[175] M. Yamashita and T. Ibaraki. Distances defined by neighborhood sequences. *Pattern Recognition Letters*, 19(3):237–246, 1986.

[176] Q. Z. Ye. The signed Euclidean distance transform and its applications. In *Proceedings 9th International Conference on Pattern Recognition*, volume 1, pages 495–499. IEEE, 1988.

[177] Y. Ye, J. Shan, S. Hao, L. Bruzzone, and Y. Qin. A local phase based invariant feature for remote sensing image matching. *ISPRS Journal of Photogrammetry and Remote Sensing*, 142:205–221, 2018.

[178] Z. Yuan, F. Li, P. Zhang, and B. Chen. Description of shape characteristics through Fourier and wavelet analysis. *Chinese Journal of Aeronautics*, 27(1): 160–168, 2014.

[179] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[180] D. Zhang and G. Lu. Shape-based image retrieval using generic Fourier descriptor. *Signal Processing: Image Communication*, 17(10):825–848, 2002.

[181] D. Zhang and G. Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1–19, 2004.

[182] D. Zhang and G. Lu. Study and evaluation of different Fourier methods for image retrieval. *Image and Vision Computing*, 23(1):33–49, 2005.

[183] H.-J. Zimmermann. Fuzzy set theory. In *Wiley Interdisciplinary Reviews: Computational Statistics*, volume 2, pages 317–332. John Wiley & Sons, Inc., 2010.

[184] S. W. Zucker and R. A. Hummel. A three-dimensional edge operator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(3):324–331, 1981.

[185] J. Žunić, P. L. Rosin, and L. Kopanja. On the orientability of shapes. *IEEE Transactions on Image Processing*, 15(11):3478–3487, 2006.