Óbudai Egyetem

Doktori (PhD) értekezés

Digitális domborzatmodellek és pontfelhők alkalmazása a terep modellezésében

Nagy Gábor József

Témavezető:
Dr. habil. Jancsó Tamás

Alkalmazott Informatikai és Alkalmazott Matematikai Doktori Iskola

Székesfehérvár, 2018. május 18.

Szigorlati bizottság:

Prof. Dr. Bakó András (elnök)
Dr. Mélykúti Gábor
Dr. Rózsa Szabolcs
Dr. José Jesús Reyes Nuñez

Nyilvános védés teljes bizottsága:

Prof. Dr. Nagy Péter (elnök)
Prof. Dr. Barsi Árpád (opponens)
Dr. Mélykúti Gábor (opponens)
Dr. Kárász Péter
Dr. Lovas Tamás
Dr. Sergyán Szabolcs (titkár)
Dr. habil. Elek István (tartalék tag)

Nyilvános védés időpontja:

Abstract

The different types of digital elevation models and point clouds are important tools in the terrain modeling. This dissertation describes and analyzes some new methods for these tools.

After the introductory chapters, the fourth chapter recommends two new tools for storing digital elevation data.

The pyramid representation is an important and popular method in the storage of image data. The lower resolution index images are very useful for displaying the image and several spatial analysis. In addition to the usual mean-based pyramid representation, I studied the minimum and maximum pyramids. This tool may be useful in several spatial processes, for example the view-shed analysis. (Thesis 1/a)

The R-tree is a very popular method for building spatial indexes, which can be used to store the element of a Triangulated Irregular Network (TIN). The 2+1 dimensional R-tree uses only the horizontal coordinates for splitting the nodes, but calculates the bounding boxes in 3 dimension. (Thesis 1/b)

The slope and the aspect are an important property of the terrain in the agriculture related geospatial analysis. The distribution of these values in a determined area can be demonstrated by different tools. A recommended diagram can show the slope and aspect in a geometrically correct and novel layout, which uses a polar coordinate system. (Thesis 2)

The elevation differences between a point of the terrain surface and the points of a circle whose center is this point may be the base of a Fourier series where the azimuth (from the examined point to a point of the circle) is the variable of the function. The coefficients of these azimuth based Fourier series are applicable for the analysis of the elevation models, for example for classification of surface points . The described analysis is implemented in easy way in varied GIS software, because the coefficients are calculated by convolutional filters. (Thesis 3)

The Fitting Disc Method is a new robust LiDAR processing method, which fits a regression plane to a point cloud in any horizontal position by fitting planes (in practice with R radius, like a disc) on it, which contains a specified portion (q) of points under the disc plane in all three sectors of the disc. This method can be used to create digital elevation models even without any filtering process. An analysis has also been described, which compares the results of the fitting disc method using different parameters (R, q) in processing of digital elevation models.

The background of Fitting Disc Method is a robust multiple linear regression with two independent variables. The Sector Based Linear Regression (SBLR) generalizes the principle of the Fitting Disc Method to any N dimensional space. (Thesis 4)

The last chapter describes a new method of processing point cloud data to interpret branches of trees. The principle of this method is based on the motion of the bubbles in a pipe, when the wall of the tube is constituted by the points of the cloud. The sphere fitting method can detect the collision of the branches and a moving sphere, and the route and the variable radius of this moving sphere represents the shape of all branches.

A simple function is recommended to generate sphere fitting values which represent the catching of a specified sphere to a point cloud by the sum of a sphere-point fitting values calculated per point, and suggests solutions for complex sphere fitting methods which takes into account the distribution of the points. More suggested methods provide the searching of sequences of fitting spheres for interpreting the branches and recognizing the shape of these branches. (Thesis 5)

The presented results can be implemented in practice, and integrated in various ways to several GIS and CAD software in the future, but this needs further research and software development work.

Tartalomjegyzék

1. fejezet

Bevezetés

Kutatásaim több, egymással összefüggő témát ölelnek fel, melyek a terep felméréséhez és informatikai eszközökkel történő kezeléséhez kapcsolódnak. A terep a Föld (vagy más égitest) felszínének és a hozzá kapcsolódó mesterséges és természetes objektumoknak az összessége; felmérése és modellezése a geodézia illetve a térinformatika legfontosabb feladata. A domborzat a terep egy lényeges eleme, a pontfelhők pedig a lézerszkenneres technológiák termékeként egyre fontosabb szerephez jutnak a domborzatnak és a terep egyéb elemeinek a térképezésében.

Értekezésemben a pontfelhőkkel és a domborzatmodellekkel egyaránt foglalkozom, bemutatva azoknak a napjaink gyakorlati alkalmazásaiban egyre jelentősebbé váló kapcsolatát. Az ismertetett eredményeim a pontfelhők és domborzatmodellek tárolásával, feldolgozásával és ábrázolásával valamint a domborzatmodelleknek és egyéb a terepet alkotó objektumoknak a pontfelhők alapján történő előállításával foglalkoznak.

1.1. A digitális domborzatmodellek jelentősége

Modellnek tekinthetünk minden olyan dolgot, ami a valóság valamilyen részletét bizonyos szempontból meghatározott pontossággal helyettesíteni képes. Ebben a dolgozatban általában olyan esetekkel foglalkozom, ahol ez a modell egy digitális adathalmaz, a valóság helyettesítése pedig az ennek az adathalmaznak a megfelelő algoritmusokkal történő feldolgozásával valósul meg.

Domborzat alatt általában a terepfelszínt mint felületet értjük. A továbbiakban, gyakorlati okokból, ezek mindig valamilyen meghatározott módon leírható felületek lesznek, amiket domborzatmodellnek nevezünk. Ha ez a modell valamilyen számítógéppel kezelhető adathalmaz, akkor digitális domborzatmodellről beszélhetünk. Mivel napjainkban a térbeli adatokat szinte már minden esetben informatikai eszközökkel kezeljük, a digitális jelzőt sokszor elhagyjuk, és egyszerűen csak domborzatmodellekről beszélünk ilyenkor is.

A domborzat a terep egyik legfontosabb jellemzője, ennek megfelelően a térképek és a térinformatikai rendszerek jelentős részében megjelenik valamilyen formában. Ha arra vagyunk kíváncsiak, hogy milyen magasságban van egy terület egy pontja vagy milyenek ott a lejtési viszonyok, hogy a víz hogyan folyik el a terep felszínén, hogy egy pont látható-e egy másik pontból, vagy hogy hány köbméter földet kell megmozgatni egy tervezett létesítmény építésekor, akkor minden esetben a domborzat egy megfelelő modelljére van szükségünk a kérdés megválaszolásához. Ez a domborzatmodell a modell fogalmának megfelelően helyettesíti a valóságbeli terepfelszínt amikor azt kell eldönteni hogy a lehullott csapadék merre folyik tovább, hogy a fény akadálytalanul el tud-e jutni egyik pontból a másikba, vagy hogy a földmunkagépeknek mekkora földtömeget kell kitermelnie.

A domborzat még azokban az esetekben is nagyon fontos kiegészítő adat lehet, amikor az alapfeladatok elvégzéséhez nincsen rá szükségünk. Erre egy jó példa amikor egy autós térképen árnyalással ábrázolják egy terület domborzati viszonyait. Ilyesmivel papír alapú autós térképeken vagy navigációs rendszerek által megjelenített térképeken egyaránt találkozhatunk. A megjelenített domborzat ilyenkor nem csupán tetszetősebbé teszi a térképet, hanem a tájékozódás során hasznosítható többletinformációt is hordoz.

Hagyományos térképi domborzatábrázolásra sokféle módszer létezik a plasztikus megjelenést biztosító különféle megoldásoktól a grafikus szerkesztésekkel a legtöbb felmerülő elemzési feladatot elvégezhetővé tévő szintvonalas ábrázolásig. Térinformatikai rendszerekben a domborzatot valamilyen a számítógép által is jól kezelhető formában kell tárolni, ami általában egymáshoz illeszkedő szabálytalan háromszög vagy négyzet alapú (vízszintes vetületű) alapelemekből felépülő felületek segítségével történik, de léteznek más megoldások is.

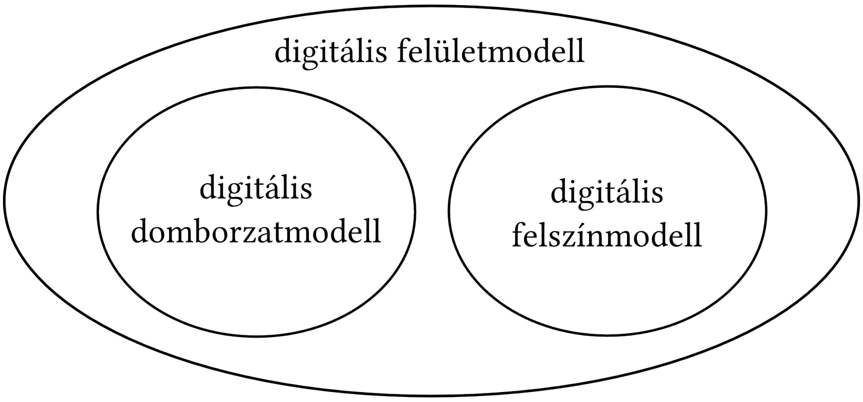
1.2. A terepfelszín értelmezésével kapcsolatos fogalmak

Gyakran használjuk a „terepfelszín”, a „fizikai földfelszín” vagy a „topográfiai földfelszín” fogalmakat. A digitális domborzatmodellek a talaj felszínét ábrázolják, amit fizikai földfelszínnek vagy topográfiai földfelszínnek is nevezünk. A digitális térkép elemei különböző fokú absztrakció eredményei, ami a domborzatmodellekre is igaz. Az absztrakció egyrészt ahhoz szükséges, hogy a terep felszínét meghatározzuk (modellezés, mit tekintünk a terep felszínének); másrészt pedig ahhoz, hogy ezt a terepfelszínt egy az alkalmazott részletességgel ábrázolható felületnek tekinthessük (generalizálás, adott részletességű modell előállítása).

Az említett absztrakció során fontos kérdés, hogy milyen objektumokat tekintünk a terepfelszín részének. A növényzetet és az építményeket nem tekintjük a terepfelszín részének, az építményekhez kapcsolódó földműveket (töltések és bevágások) viszont sokszor már igen. További kérdések merülnek fel a vízfelületeknél, hogy azok a részletes mederrel vagy egy közepes vízszinthez tartozó sík felülettel jelenjenek meg a modellen. A választást ebben az esetben általában a rendelkezésünkre álló adat határozza meg.

Létezik még digitális felszínmodell is, ami a terepnek és a rajta elhelyezkedő természetes és mesterséges tereptárgyaknak a felülről látható felszínét határozza meg. Jelentősége abban áll, hogy fotogrammetriai technológiák segítségével ezt a felszínmodellt tudjuk közvetlenül meghatározni, valamint hogy digitális ortofotó előállításakor vagy térhatású megjelenítésekkor is erre a modellre van szükségünk.

A digitális domborzatmodell és a digitális felszínmodell csak az ábrázolt felület tekintetében különböznek egymástól, tárolásuk és kezelésük azonos eszközökkel, azonos elven történik. Amikor digitális felületmodellekről beszélünk, akkor az lehet dombor-



1.2.1. ábra. A „digitális felületmodell”, a „digitális domborzatmodell” és a „digitális felszínmodell” fogalmak kapcsolatának szemléltetése Venn-diagram segítségével.

zatmodell, felszínmodell vagy bármilyen más, akár képzetes felületnek a modellje. (1.2.1 ábra)

1.3. A digitális felületmodellek értelmezése

Egy felületmodell tekinthető egy $f(x, y)$ kétdimenziós függvénynek, amely a vízszintes koordinátákhoz rendeli a felület magasságát ($f : \mathbb{R}^2 \rightarrow \mathbb{R}$). Az az előnyös ebben a megközelítésben, hogy a függvénynek tekintett felület számos jellemzőjét definiálhatjuk így a matematikai analízis közismert eszközeivel, például a függvény különféle deriváltjaira hivatkozva. A felületmodell kétdimenziós függvényként való megközelítése azt jelenti, hogy a vízszintes koordináták ismeretében a modell alapján meg tudjuk határozni a magasságot.

Felületmodell alatt általában egy a felületet valamilyen módon leíró adathalmazt értünk. Ilyenkor információként tekintünk a felületre. A felületmodellt ekkor egy olyan összetett adatként határozhatjuk meg, amelyikből a megfelelő metódusok segítségével megmondható a felület magassága (és esetleg egyéb jellemzői) egy helyen. Ez a megközelítés objektum szemléletű, mivel az adatok és a velük végezhető műveletek egységére épül, ugyanakkor a vízszintes koordinátákból a magasságot meghatározó metódussal a függvényszerű megközelítéshez jutunk.

A digitális felületmodellt sokféle adatszerkezettel meg lehet valósítani. A legelterjedtebb megoldások a szabályos négyzetrácshálón és a szabálytalan háromszöghálón alapulnak, az értekezésben is főként ezekre fogok koncentrálni. Az egyéb lehetőségek közül számos további megoldást ismertet a [88].

1.4. A pontfelhők szerepe a terep modellezésében

A terep felmérésének egy korszerű és egyre inkább elterjedő technológiája a lézerszkennelés. A technológia lényege, hogy nagy mennyiségű (akár másodpercenként több százezer vagy millió) lézeres távolságmérést végzünk meghatározott irányokba, letapogatva ezáltal a körülöttünk (légi lézerszkenner esetében alattunk) lévő objektumokat. A mérések történhetnek fix álláspontokról (földi lézerszkennelés, angolul Terrestrial Laser Scanning, rövidítve TLS) vagy mozgó járműről (mobil vagy légi lézerszkennelés).

A mérések eredményeképpen sok millió pontot kapunk, amelyek összességét pontfelhőnek (point cloud) nevezzük. A pontfelhő egyes pontjainak információtartalma önmagában jelentéktelen, lényegében csak annyi ismeretet hordoz, hogy az adott pontban valamiről visszaverődött a távméréshez használt lézerfény. (Pontosabban annyit, hogy a mérés megbízhatóságától függő valószínűséggel van valami a pontnak a mérés pontosságától függő kiterjedésű környezetében.) Ez eltér a klasszikus geodéziai felmérés során megszokottól, amikor minden felmért pontnak jól meghatározott szerepe van a létrehozandó térképen; általában valamilyen objektum egy alakjelző pontját jelentik, vagy legalább a terepfelszínnek egy jellemző pontját.

Bár egyetlen pontja önmagában nem mond szinte semmit, a teljes pontfelhő a felmért objektumok (objektum itt most lehet bármi, akár a terepfelszín is) rendkívül részletes leképezését adja, ami a későbbiekben akár a felmérési munka eredeti céljain túlmutatva is számos további információ kinyerését teheti lehetővé. Ebből adódóan lézerszkenneres felmérésekkel előállított pontfelhők akár alapadat jelleggel is készülhetnek. Erre nemzetközi szinten is úttörő példa a BKK Közút „Közúti Adatgyűjtő Rendszer” (rövidítve: KARESZ[1]) nevű rendszere, amelynek alapja Budapest úthálózatának lézerszkenneres mérésekkel előállított pontfelhője [84]. Szlovéniában az ország teljes területéről készültek LiDAR felmérések, amelyek ráadásul ingyenesen letölthetőek[2].

A terepi felmérés a lézerszkenneres technológiák alkalmazásával viszonylag rövid idő alatt elvégezhető, hiszen a modell létrehozásának időigényes folyamata már irodai munka, ami így független az időjárási körülményektől, illetve egyéb a terepen fellépő akadályozó tényezőktől (pl. a munkaterületen folyó építési munkák).

A lézerszkenneres felmérés a fentiek alapján sok szempontból hasonlít a fotogrammetriai felmérésre: a munka nagy része a felmérés után, a pontfelhő feldolgozásakor jelentkezik. A térképezéshez kapcsolódó modellezés és az ehhez szükséges absztrakció is ekkor történik, ellentétben a geodéziai felméréssel, amikor ez már a terepen megkezdődik.

Bár a pontfelhők a felmért terep nagyon részletes (nagy információtartalmú) és geometriai szempontból is helyes reprezentációját nyújtják, térinformatikai elemzések alapjául közvetlenül nem használhatóak. Mindenféleképpen szükség van még valamilyen feldolgozott termék előállítására a pontfelhő alapján az ilyen műveletek előtt. Ebből a szempontból a pontfelhők a digitális ortofotókkal mutatnak hasonlóságot.

A pontfelhők feldolgozásának támogatása, az objektumok kiértékelésének részleges vagy teljes automatizálása fontos kutatási terület, amelynek eredményei az egész technológia hatékonyságát és termelékenységét jelentősen befolyásolják. A légi lézerszkenneléssel nyert pontfelhők egyik legfontosabb felhasználási területe a digitális felszínmodellek és a digitális domborzatmodellek előállítása.

[1]A KARESZ-ról bővebben a http://budapestkozut.hu/szakfelugyelet címen elérhető előadásban lehet olvasni.

[2]A http://gis.arso.gov.si/geoportal/catalog/main/home.page oldalról.

1.5. Az értekezésben alkalmazott jelölések és kifejezések

Az értekezésben előforduló mennyiségekre és jellemzőkre igyekeztem egységes jelöléseket alkalmazni. A legtöbb esetben a Magyarországon szokásos, az EOV által is használt geodéziai koordináta-rendszer helyett matematikai koordináta-rendszert használok, melynek x tengelye a keleti, az y tengelye pedig az északi irányba mutat. Az irányszöget ilyenkor is az y tengelytől (északi irány) kiindulva, az óramutató járásának megfelelő irányban értelmeztem. (Tehát az északkeleti tájolás és az irányszög értelmezése ekkor is a megszokott marad, csak a két koordináta jelölése cserélődik meg.) Erre azért volt szükség, mert nagyon sok esetben kezelem a modellezett felületeket kétdimenziós függvényként, és alkalmazom rajtuk a matematikai analízis eszközeit, ami nagyon zavaró lenne a matematikában megszokottól eltérő jelölésekkel.

Az értekezés szövegében a pont helyett a pozíció kifejezést használom olyankor, amikor a térnek (vagy a síknak) egy tetszőleges pontjáról van szó, nem pedig egy pontfelhő egy pontjáról, egy domborzatmodell egy támpontjáról vagy bármilyen más objektumnak valamilyen alakjelző pontjáról. Tehát pont alatt egy pont objektumot vagy egy összetettebb objektumnak valamilyen alapelemét értem, pozíció alatt pedig egy tetszőleges térbeli helyet.

A felhasznált szakirodalmat a dolgozat végén található irodalomjegyzékben foglaltam össze, előtte elhelyezve a saját publikációim jegyzékét, ahol elkülönítve soroltam fel a téziseket megalapozó publikációkat és az egyéb közleményeket. A dolgozat szövegében ezeknek a jegyzékeknek a sorszámait szögletes zárójelek közé téve hivatkozok az egyes művekre. A tudományos szakirodalomnak nem tekinthető forrásokra (pl. Interneten elérhető anyagok, szoftverek és műszerek dokumentációi, szabványok) lábjegyzetekkel hivatkozom.

1.6. Köszönetnyilvánítások

Munkahelyemen, az Óbudai Egyetem Alba Regia Műszaki Karának Geoinformatikai Intézetében, illetve annak jogelődjében a Nyugat-magyarországi Egyetem Geoinformatikai Karán 2005 óta dolgozom. A dolgozatban bemutatott kutatások közül néhányat az ott rendelkezésemre álló eszközök segítségével tudtam elvégezni.

Jelenlegi doktori cselekményeimet megelőzően a Nyugat-magyarországi Egyetem Erdőmérnöki Karán működő Kitaibel Pál Környezettudományi Doktori Iskola Geokörnyezettudományi Programjának PhD hallgatója voltam. Témám címe a „Digitális domborzatmodellek alkalmazása a környezeti hatásvizsgálatokban”, témavezetőm prof. dr. Márkus Béla volt. Értekezésemet nem készítettem el, de az akkori kutatásaim több eredményét is be tudtam építeni a mostani dolgozatomba.

Sok köszönettel tartozom témavezetőmnek dr. habil. Jancsó Tamásnak az értekezés és a kapcsolódó publikációk elkészítésében nyújtott segítségéért, és még korábban az NymE GEO kutatási dékánhelyetteseként nyújtott támogatásáért. Kollégáim közül ki kell még emelnem dr. habil. Földváry Lóránt kutatási dékánhelyettest, valamint dr. Busics György intézetigazgatót és dr. habil. Györök György dékánt, akik folyamatosan buzdítottak doktori dolgozatom és a hozzá kapcsolódó publikációs tevékenységek minél gyorsabb befejezésére.

Az ilyenkor szokásosnál is több köszönettel tartozom családomnak. Szüleim, Nagy József Ödön és Nagyné Csontos Gyöngyi a 7. fejezetben bemutatott, a pontfelhő alapján létrehozott domborzatmodellek minőségének vizsgálatához használt geodéziai mérések elvégzésében nyújtottak komoly segítséget; míg menyasszonyom, dr. Ungvári Zsuzsanna több ábra elkészítésében is közreműködött és számos esetben tudta szakmai szempontból is segíteni a munkámat.

Végül szeretném megköszönni mindazoknak a munkáját, akik opponensként vagy valamelyik bizottság tagjaként működtek közre a fokozatszerzési eljárásomban; és hasznos észrevételeikkel járultak hozzá ehhez a dolgozathoz.

2. fejezet

A digitális domborzatmodellekkel kapcsolatos alapvető ismeretek

Digitális domborzatmodellnek egy olyan adathalmaz lehet alkalmas, amiből egy meghatározott területre eső vízszintes koordinátáival megadott pozícióhoz le tudunk vezetni egy magasságot és további szükséges jellemzőket, illetve megfelelő algoritmusok segítségével el tudjuk végezni a számunkra szükséges egyéb műveleteket. A fenti feltételnek sokféle adatmodell megfelel, amelyek alkalmazása különféle esetekben lehet hatékony.

A digitális domborzatmodellekkel kapcsolatban nagyon sokféle elemzés illetve egyéb adatfeldolgozási eljárás ismert [56, 88, 94]. Ezek között a különböző összetettségű műveletek között vannak amelyek a domborzatmodellnek az előállítását szolgálják, míg mások különféle információknak a domborzatmodellből való kinyerésére használhatóak.

A domborzattal kapcsolatos elemzéseket többféle módon is osztályozhatjuk a felhasználás célja alapján, a felosztás szubjektív jellege miatt. A domborzatmodellekkel végzett elemzések egy vagy több a domborzatmodellen végrehajtott műveletből épülhetnek fel.

A továbbiakban egyszerű műveleteknek fogom nevezni azokat a domborzatmodellekkel kapcsolatos műveleteket, amikor a domborzatmodell egy pontjának valamilyen jellemzőjét állapítjuk meg a kérdéses pozíció szűkebb környezetének a vizsgálatával. Ezeknek az egyszerű műveleteknek az eredménye természetesen nem csak egyetlen, egyszerű adat lehet, hanem akár egy raszter állomány is, amennyiben a rácsháló minden pontjában meghatározzuk az adott értéket. A számított értékek általában számok, de másféle eredmény is elképzelhető, például logikai érték vagy valamilyen kategóriába sorolás.

Összetett műveleteknek azokat a domborzatmodellel végezhető műveleteket fogom nevezni, amelyek nem definiálhatóak csupán egyes pontoknak és szűkebb környezetüknek az egymástól független vizsgálatával. Ezeknek a műveleteknek a hátterében összetettebb, a domborzat egészét érintő összefüggések vannak.

2.1. Digitális domborzatmodell előállítása

A digitális domborzatmodelleket többféle technológiával is elő lehet állítani, amelyek a költségek, a felbontás és a pontosság tekintetében is sokfélék lehetnek. Az alábbiakban a legfontosabb módszereket foglalom össze és mutatom be röviden.

2.1.1. Hagyományos, geodéziai és topográfiai felmérések

A domborzat felmérése történhet olyan módon, hogy a domborzat meghatározott pontjait a felmérést végző személy (vagy annak segédje, figuránsa) a terepen egyenként felkeresi, majd megfelelő eszközökkel (mérőasztal, mérőállomás, GNSS technológia) meghatározza a pont térbeli helyzetét (vízszintes helyzet és magasság). Ezeknek a módszereknek a hátránya, hogy drágák, termelékenységük még legkorszerűbb eszközöket használva sem közelíti meg a többi technológiáét, viszont megbízhatóságuk és a létrehozott domborzatmodell minősége ilyenkor a legjobb, hiszen a felmérést végző személyesen járja be a felmérendő terep minden részletét. Akár egyéb objektumok felmérésével együtt is végezhető.

A felméréskor a terep tetszőleges pontjának meghatározására lehetőségünk nyílik, így módunk van a domborzat jellegzetes elemeit, az idomvonalak jellemző pontjait felmérve kevesebb ponttal is részletesen meghatározni a terepfelszínt.

2.1.2. Fotogrammetriai felmérés

Sztereofotogrammetriai módszerekkel a repülőgépről lefényképezett terepnek egy képpárból előállított térbeli modelljét használva is lehetőség nyílik a domborzat kiértékelésére. Régebben az analóg technológiát használó műszerekkel közvetlenül a szintvonalakat rajzolták meg. A meghatározott magasságra beállított mérőjelet ilyenkor úgy mozgatták vízszintes értelemben, hogy az közben a terep felszínén maradjon. Egy másik elterjedt megoldás az volt, hogy az adott irányban automatikusan mozgatott mérőjelet a magasság változtatásával a terepen tartva határozták meg a terepfelszín adott irányú metszeteit. Ez utóbbi módszert elsősorban az ortofotók előállításához kapcsolódóan használták. Az analóg és analitikus fotogrammetriai eszközöket használva sokféle grafikus vagy számszerű adatrögzítésre nyílt lehetőség a domborzatik adatok kiértékelésekor.

Fotogrammetriai felmérések digitális kiértékelésekor a felszínmodell előállítása szinte teljesen automatizált. A megfelelő feldolgozószoftverek a képpárok átfedő részeinek automatikus illesztésével állítanak elő nagy mennyiségű támpontot a létrehozandó domborzatmodell számára. Ha szükség van rá, akkor később a digitális ortofotó előállítása is az így létrehozott modellt felhasználva történik.

Fontos megemlíteni, hogy a bemutatott fotogrammetriai technológiák közvetlen eredménye nem a domborzatnak, hanem a repülőgépről lefényképezhető felszínnek a modellje, vagyis digitális domborzatmodell helyett digitális felszínmodellt kapunk. Az erdős vagy egyéb növényzettel benőtt területeknél ezért a növényzet átlagos magasságának ismeretében korrekciókat kell alkalmaznunk, illetve ki kell hagyni a domborzatmodellből az épületek és más mesterséges létesítmények felületeit.

2.1.3. Lézerszkenneres felmérés

A domborzat egy kisebb, néhányszor száz négyzetméteres darabját akár földi lézerszkenneres technológiával is fel lehet mérni. Egy ilyen megoldással egy korlátozott méretű terület rendkívül részletes domborzatmodelljét lehet elkészíteni, ami alkalmas a domborzat legapróbb megfigyelhető részleteinek a felmérésére, illetve a mérést a későbbiekben megismételve lehetőségünk nyílik az időbeli változások tanulmányozására is.

Nagy kiterjedésű terület felmérésére a légi lézerszkenneres (LiDAR) technológiák alkalmasak. Ezen a területen az egyik legfrissebb újdonság a LiDAR felmérésekre alkalmas drónok megjelenése[1], ami a jövőben várhatóan egyszerűbbé teszi az ilyen jellegű adatok előállítását kisebb kiterjedésű területek esetében is.

A pontfelhők ilyen célú kiértékelésével a 7 fejezetben foglalkozom részletesen.

2.1.4. Egyéb felmérési technológiák, adatforrások

A Shuttle Radar Topography Mission (SRTM) keretében 2000 februárjában 11 napon keresztül végeztek radaros méréseket az Endeavour űrsikló fedélzetéről. A mérések eredményeként egy globális (a déli szélesség 57. foka és az északi szélesség 60. foka közötti szárazföldek területeit tartalmazó) domborzatmodellt készítettek. A WGS84 ellipszoidon előállított, foknégyszögenként letölthető domborzatmodellek felbontása 3 illetve 1 szögmásodperc. Ez utóbbi, nagyobb felbontású modell sokáig csak az Egyesült Államok területére volt elérhető.

Fontos, a kutatásaim és értekezésem írása közben bekövetkezett fejlemény az SRTM-el kapcsolatban, hogy a másodperces felbontású modellek már nem csak az USA területén állnak ingyenesen bárki rendelkezésére. Ennek következtében több bemutatott elemzést eredetileg a három másodperces modell adataival végeztem el, majd esetenként az elemzést megismételtem az egy másodperces modell adataival is.

Az SRTM modellből nem csupán a sarkvidéki területek hiányoznak, hanem mindenfelé találhatóak benne ismeretlen (NULL) magasságú rácspontok. Ezek az üresen hagyott elemek elsősorban vízfelületeken illetve magasabb hegységek mélyebb völgyeiben találhatóak. A modellből elérhető olyan változat is, ahol ezeket a helyeket a környező ismert magasságú rácspontok alapján interpolált értékekkel töltötték ki. Az SRTM modellel kapcsolatos elemzések és ismeretek olvashatók a [78, 134, 112]-ben, magyar nyelven pedig a [130]-ban.

Az ASTER GDEM domborzatmodellt a Terra műhold főleg infravörös tartományban működő ASTER szenzorának 15 méteres felbontású felvételeinek sztereofotogrammetriai feldolgozásával állították elő, felbontása 1 szögmásodperc. A munkához a 2000 és 2009 között készített felvételeket használták fel, a feldolgozás nagyon magas fokú automatizálással, minimalizált emberi munkaidő-ráfordítással zajlott. Nem csupán a felületmodellek képpárok alapján történő kiértékelését végezték teljesen automatikusan, hanem például a felvételek felhős részeinek kimaszkolását is. Az ASTER bemutató elemzése a [126]-ben olvasható.

Az ASTER minősége a fent bemutatott okokból a nagyobb felbontás ellenére sem éri el a három másodperces SRTM-ét, sokkal több helyen fordulnak elő benne durva hibák. További probléma lehet az ASTER-el, hogy az adatgyűjtés módjából adódóan nem a topográfiai földfelszínt, hanem a látható földfelszínt ábrázolja, vagyis nem domborzatmodell, hanem felszínmodell.

Az SRTM és az ASTER GDEM modellek ingyenes elérhetőségük és globális kiterjedésük miatt népszerű adatforrások a legkülönfélébb térinformatikai rendszerekben. A konkrét alkalmazás igényeinek ismeretében, a két adatforrás tulajdonságait szem előtt

[1] A Riegl például 2015-ben illetve 2016-ban jelent meg VQ-480-U és VUX-1UAV termékeivel, amelyekről bővebben a http://www.riegl.com/products/unmanned-scanning/ címen lehet olvasni.

tartva érdemes választani közülük. Sokféle ingyenesen használható adatforrás részletes bemutatása található a [128]-ben.

A globális domborzatmodellek sorában a legújabb a TanDEM-X. Ez a Német Űrügynökség[2] és a EADS Astrium[3] együttműködésével készült. A mérés elve hasonlít az SRTM-éhez, de itt nem egy űrsiklón és az arról kinyújtott karon vannak elhelyezve a műszerek, hanem a két egymáshoz közel (250-500 méter) keringő műholdon, amelyek az értekezés írásának idején már közel 8 éve működnek. A hosszabb bázis és a sokkal több mérési idő pontosabb és részletesebb domborzatmodell készítését tette lehetővé. A 12 méteres felbontású és 2 méteres pontosságú modellekből ingyenesen csak néhány mintaterületet lehet letölteni. A folyamatosan működő műholdak lehetővé teszik a domborzat változásainak vizsgálatát is.

2.1.5. Másodlagos adatgyűjtés, régebbi adatok felhasználása

A térbeli adatok létrehozásának az a legegyszerűbb és legolcsóbb módja, ha korábbi felmérések adataiból indulunk ki. A legegyszerűbben a végtermékként létrehozott térképekhez lehet hozzáférni, de sokszor célszerű lehet más, a felmérés során létrejött munkarészeknek a felhasználása is. Mivel a domborzat a térképek egyik legkevésbé változó eleme, a korábban felmért állapot módosulásával nem igazán kell számolni, ezért csupán a felhasználni kívánt korábbi felmérés minőségi jellemzői határozzák meg az így kapható eredményt.

A hagyományos térképeken a domborzat grafikus ábrázolására többféle módszert is alkalmaztak, melyek közül műszaki szempontból a szintvonalas ábrázolás a jelentős. A domborzat szintvonalas térképéből kiindulva, hagyományos grafikus szerkesztésekkel (körző, vonalzó) szinte mindenféle műszaki szempontból fontos feladatot el lehet végezni (pont magasságának és egyéb jellemzőinek meghatározása, metszetek készítése tetszőleges síkok mentén, rézsűk és bevágások szerkesztése, földtömegszámítások, a terepfelszín és egy tetszőleges egyenes döféspontjainak meghatározása, semleges vonalak szerkesztése, stb.). Ezeken a lehetőségeken túl az ebben némi gyakorlattal rendelkező személyek a szintvonalas térképre tekintve nagyon jól el tudják képzelni az ábrázolt terület domborzati viszonyait.

Az előbbiek után kézenfekvő megoldásnak tűnhetne a domborzat digitális modellezése a szintvonalas térkép digitális megfelelőjének, a digitális szintvonalmodellnek a segítségével; de a grafikus térképekkel ellentétben egy digitális térképen a szintvonalakkal komoly problémák merülnek fel, ha a megjelenítésen túl más célokra is használni szeretnénk őket, például ha meg akarunk határozni egy vízszintes pozícióhoz tartozó magasságot. A szintvonalas térkép egy jó példa arra, amikor a számítógép számára bonyolultak az emberi agy által egyszerűen megoldható feladatok. Elvileg akár captcha-ként is lehetne alkalmazni egy szintvonalas térképet (annak raszterizált képét), amely alapján egy azon megjelölt pozíció magasságát kellene a felhasználónak meghatározni, bizonyítandó azt, hogy nem egy másik számítógép van a kapcsolat túlsó végén.

Bár a digitális domborzatmodellek teljesen más elvek szerint épülnek fel, a szintvonalas térképek többféle formában is előfordulnak térinformatikai rendszerekben. Egyrészt

[2] German Aerospace Center, németül Deutsches Zentrum für Luft- und Raumfahrt vagy röviden DLR

[3] Az European Aeronautic Defence and Space Company, közismertebb nevén az Airbus csoport tagja.

régebbi, papír alapú térképek szintvonalainak digitalizálásával keletkezett adatként, melyet a későbbiekben általában valamilyen a számítógép által már könnyen kezelhető domborzatmodellé alakíthatunk; bár ezt a fajta adatforrást sokan ellenjavallják, a gyakorlatban mégis sokszor előfordul, mert a domborzati adatok általában ilyen formában állnak rendelkezésre a régebben készült hagyományos térképeken. A másik esetben a grafikus megjelenítéshez használunk szintvonalakat, mert a felhasználók megszokták és igénylik ezt a fajta domborzatábrázolást; így ilyenkor a szintvonalas térképet valamilyen digitális domborzatmodellből levezetve állítjuk elő. (Bővebb példa erre a 2.5.4 ábrán látható.)

A szintvonalakból származtatott digitális domborzatmodellekkel kapcsolatban sok fenntartással és negatív példával találkozhatunk. Kifejezetten rossz az a megoldás, amikor csak a szintvonalak töréspontjait használják fel a domborzatmodell készítéséhez, eldobva ezzel azt az információt, hogy az nem csupán egy pontsorozat, hanem a rá illeszkedő vonal valamennyi pontja része a terep felszínének. Egyes domborzati formák helyes és pontos leképezéséhez fontos lehet még, hogy domborzatnak a hagyományos térképeken kótált pontként megjelenő egyes pontjait (pl. hegyek csúcsai) is felhasználjuk, lehetőség szerint azt is figyelembe véve, hogy a felület érintője vízszintes ezekben a pontokban.

Szintvonalak alapján történő domborzatmodell létrehozással foglalkozik a [11].

A másodlagos adatgyűjtés nem csupán a szintvonalas adatok feldolgozását jelentheti. Ide lehet sorolni minden olyan eljárást, ami korábban végzett terepi mérések feldolgozásán alapul. Ezek egy része csak abban különbözik az elsődleges adatgyűjtéstől, hogy a feldolgozott terepi mérések (a terepi mérést itt most tágan kell értelmezni, fotogrammetriai és távérzéskelési eljárásokat is ide lehet sorolni) feldolgozása sok évvel a mérés után történik, az eredetinél korszerűbb technológiával.

Vannak olyan esetek, amikor nem az eredeti mérési eredmény vagy a végtermék, hanem a felmérési technológia valamilyen köztes munkarésze alapján hozunk létre digitális domborzatmodellt. Ilyen lehet például, amikor egy topográfiai felmérés alaplapján megrajzolt idomvonalakat digitalizálunk, és azok alapján hozunk létre valamilyen modellt, vagy ha egy négyzethálós területszintezés eredményeként a rácspontokban kapott magasságok bevitelével egy GRID modellt hozunk létre értelemszerű módon.

A jövőben várhatóan az adatgyűjtési technológiák fejlődésével, ahogy egyre egyszerűbbé és olcsóbbá válik kisebb kiterjedésű területek felmérése UAV platformról fotogrammetriai vagy LiDAR technológiákkal, illetve nagyobb kiterjedésű területekre egyre részletesebb és jobb kész domborzatmodellek lesznek alapadatként érhetőek, egyre kisebb jelentősége lesz a régebbi, nem digitális formában rendelkezésre álló domborzati adatok felhasználásának.

2.2. GRID modellek

A domborzat modellezésének egy egyszerű módszere az, ha a terepfelszín magasságait egy szabályos rácsháló pontjaiban adjuk meg. Egy tetszőleges pont magasságát meghatározhatjuk interpolációval a környezetében lévő rácspontok magasságából kiindulva. Az ilyen domborzatmodelleket GRID modellnek hívjuk.

A GRID modellt úgy is tekinthetjük, mint a felszínt leíró függvénynek és egy τ rácsállandójú Dirac-impulzus sorozatnak a szorzatát. Ebből az összefüggésből is levezethető, hogy a modell nem képes kimutatni a domborzat 2τ-nál kisebb kiterjedésű részleteit. [28, 76, 77]

A GRID modell az adattárolás szempontjából a raszter képekkel azonos elven működik, hiszen mindkét esetben egy kétdimenziós tömböt (egy mátrixot) kell a számítógépnek kezelnie. A módszer előnye, hogy egy vízszintes koordinátapárból a 2.2.3 összefüggés segítségével nagyon egyszerűen és gyorsan meg lehet mondani, hogy melyik rácspontokra van szükségünk az interpolációhoz. Az ehhez szükséges lépések száma a rácspontok számától függetlenül konstans, vagyis a művelet $O(1)$ idő alatt elvégezhető. Hátránya, hogy általában nagy a tárigénye, és a konstans rácstávolság miatt nem tud alkalmazkodni a terep különféle részletességű ábrázolást igénylő területeihez.

2.2.1. A GRID modellek tárolásának általános kérdései

Mivel a GRID modellekben tárolt adat logikai felépítése a raszter képek adatával teljesen megegyezik, a tárolásuk fizikai modellje is hasonló. A gyakorlatban ez minden olyan képformátum alkalmazhatóságát jelenti, ami az egyes raszterek (pixelek) értékeiként lehetővé teszi a domborzatmodell magasságainak megadására alkalmas értékek tárolását.

A konkrét magasságokat leíró számokon túl szükség lehet egy a nem ismert magasságú rácspontokhoz rendelhető értékre is, amivel azt fejezzük ki, hogy az adott rácspontban nem ismerjük (megfelelő pontossággal) a terep magasságát. Fontos, hogy a későbbiekben ezt a domborzatmodellen végzett műveletek során is megfelelően kezeljük, ne végezzük el a kívánt számításokat ezzel a teljesen mást jelölő értékkel, hanem az ilyen adatok felhasználásával meghatározandó értékek a számítás eredményében is ismeretlen adatok legyenek.

Sok képformátumnál gondot okozhat, hogy a pixeljei csak 0 és 255 közötti értékeket vehetnek fel, ami a domborzatmodellek tárolásához nem elég. A TIFF (Tag Image File Format) formátum[4] használatakor lehetőségünk van lebegőpontos számoknak vagy 8 bitesnél hosszabb egész számoknak a használatára is, ami már alkalmas lehet digitális domborzatmodellek kezelésére. A formátum további hasznos tulajdonsága, hogy egy TIFF állományban tetszőleges számú réteg elhelyezhető, bár ez főként multispektrális adatok tárolásánál hasznos.

Térinformatikai célokra a GeoTIFF formátumot[5] szokás használni, ami egy olyan szabályos TIFF állomány, ahol kihasználják a formátum névadó tulajdonságát, miszerint a tárolt raszter adatokhoz címkeszerűen, kulcs-érték párokkal további metaadat-jellegű ismereteket lehet fűzni; és ilyen módon adnak meg többek között olyan dolgokat mint például az alkalmazott vetületi rendszer meghatározásához vagy a rácshálónak ennek a vetületnek a koordináta-rendszerében való elhelyezéséhez szükséges ismeretek.

Egy nagyon egyszerű megoldás lehet a GRID modellek (és mindenféle raszter adat) kezelésére az, amikor az értékek kétdimenziós tömbjét sor- vagy oszlop-folytonosan kiírjuk egy bináris állományba. A rács méreteinek (sorok és oszlopok száma) valamint az alkalmazott számtípus hosszának ismeretében könnyedén meg lehet mondani, hogy egy adott rácspont magasságát leíró adat az adathalmaz (a fájl) melyik részén található. Az

[4] A TIFF formátum részletes leírása az ISO 12639:1998 szabványban található meg.

[5] A GeoTIFF formátumról bővebb információt a http://trac.osgeo.org/geotiff/ oldalon lehet találni.

ilyen fajta megoldásokat nyers (raw) bináris adatnak is szokás nevezni. A nyers jelző arra vonatkozik, hogy semmiféle tömörítési vagy indexelési lehetőséget nem alkalmazunk.

Adatcsere céljára nagyon jól használható az ArcInfo program szöveges alapú ASCII GRID[6] formátuma, ami egy fejlécet követően tartalmazza az egyes rácspontok magasságait. Bár a szöveges formátum mérete nagyobb még a nyers bináris formátumokénál is, de sok esetben mégis jól használható, főleg egyszerűbb, saját fejlesztésű programok kimeneti formátumaként. A fejlécben tetszőleges érték meghatározható az ismeretlen adatokat jelző értékként, ami nem a számszerűen hozzá tartozó, hanem az ismeretlen értékeket (domborzat esetében az ismeretlen magasságokat) fogja jelenteni.

2.2.2. A GRID modellek georeferálásának kérdései

A GRID modell alkalmazásakor a rácspontok magasságai mellett meg kell még valahogyan határozni a rácsháló elhelyezkedését is. Ez jelenti egyrészt a rácsháló elhelyezkedését egy koordináta-rendszerben, másrészt ennek a koordináta-rendszernek a viszonyát a Föld felszínéhez képest.

A rácsháló elhelyezkedése megadható az egyik (formátumtól illetve alkalmazástól függ, hogy melyik) sarokpontjának a koordinátáival, valamint a rácsháló vonalainak távolságát és a koordináta-rendszer tengelyeihez képesti irányát kifejező adatokkal. Általános esetben a rácsháló soraihoz és oszlopaihoz kapcsolódóan is meg lehet adni egy tetszőleges vektort, ami az egy oszloppal vagy egy sorral való elmozdulást jelenti az alkalmazott koordináta-rendszerben. Ez tulajdonképpen egy tetszőleges affin transzformációt jelent a rácsháló koordináta-rendszere és az alkalmazott vonatkozási rendszer koordináta-rendszere között.

Többféle megkötés is elképzelhető a rácsháló elhelyezkedésével kapcsolatban. A rácsháló lehet a koordináta-rendszer tengelyeivel csak párhuzamosan elhelyezhető, vagy azokhoz képest egy egységes (soronként és oszloponként külön nem szabályozható) szöggel elforgatható. A rácsháló felbontásával kapcsolatban is elképzelhető egyes alkalmazások és formátumok esetében, hogy egységesnek kell lennie mindkét irányban.

A vonatkozási rendszer az alapfelület és az azon elhelyezett vetület tulajdonságainak teljeskörű leírásával vagy egy kódszámmal is megadható. A kódszám tárolása esetén egy adatbázisból lehet kikeresni a megadott vonatkozási rendszer leírását. A különféle vonatkozási rendszerek sorszámmal való azonosításához főként az EPSG[7] rendszert szokás használni. Ebben a rendszerben az EOV a 23700, a WGS84 ellipszoid a 4326 kódszámmal érhetőek el, a Magyarország területére eső 33-as és 34-es számú UTM vetületek azonosító számai pedig 32633 és 32634.

Különféle térinformatikai alkalmazásokban általában elég az EPSG sorszámokat megadni, a vonatkozási rendszer paramétereit (a vetület típusa és elhelyezkedése az alapfelületen, valamint az alapfelület elhelyezkedése egy meghatározott referencia-rendszerhez, általában a WGS84-hez képest) egy a szoftver részét képező adatbázisból kérdezi le a

[6] Az ArcINFO ASCII GRID formátumáról a http://resources.esri.com/help/9.3/arcgisdesktop/com/gp_toolref/spatial_analyst_tools/esri_ascii_raster_format.htm vagy a https://en.wikipedia.org/wiki/Esri_grid oldalon található részletes információ.

[7] Az EPSG az European Petroleum Survey Group, vagyis egy kőolajipari nemzetközi szervezet nevének rövidítése. Ez a szervezet adta ki először azt a vonatkozási rendszerek adatait tartalmazó adatbázist, ami jelenleg a http://spatialreference.org/ címen érhető el.

program. Fontos tudni, hogy az ezekben az adatbázisokban elérhető leírásokkal általában nem érhető el geodéziai pontosság. A geodéziai igényeket is kielégítő pontosság ún. javító rácsokkal érhető el, amelyek a klasszikus vetületi átszámítások után maradó eltéréseket tartalmazó raszter állományok[8].

A GRID modell elhelyezkedését az alkalmazott koordináta-rendszerben célszerű egy olyan affin transzformációval megadni, ami a rácsháló által kijelölt rendszer (a rácsháló sorai és oszlopai) és az alkalmazott koordináta-rendszer kapcsolatát határozza meg. Ez hat paramétert jelent, ami közül kettő a rácsháló kezdőpontjának pozícióját (X_0, Y_0) adja meg az alkalmazott koordináta-rendszerben (feltéve hogy a rácsháló sorainak és oszlopainak indexelését nullától kezdjük), két-két további paraméter pedig a rácsháló egymást követő soraihoz illetve oszlopaihoz tartozó eltolások vektorait határozza meg.

Egy rácsháló egy pontjának koordinátáit ezeknek a paramétereknek az ismeretében az affin transzformáció képletével egyszerűen ki lehet számítani:

$$\begin{aligned} X &= X_0 + T_1 \cdot o + T_2 \cdot s \\ Y &= Y_0 + T_3 \cdot o + T_4 \cdot s \end{aligned} \tag{2.2.1}$$

ahol s és o a rácspont sorának és oszlopának a sorszámát jelentik. Ezek tekinthetőek a rácsháló magasságait tároló kétdimenziós tömb indexeinek is. A $[T_1, T_3]$ vektor a rácsháló egymást követő szomszédos oszlopaihoz a $[T_2, T_4]$ vektor pedig a rácsháló egymást követő szomszédos soraihoz tartozó eltolásokat adják meg. Ha a rácsháló az alkalmazott koordináta-rendszerhez képest nincsen elforgatva, akkor a T_2 és a T_3 paraméterek értéke nulla, a T_1 és a T_4 pedig az oszlopok illetve a sorok távolságával egyeznek meg. A T_4 értékét általában negatív számként adják meg, mivel a rácsháló sorait északról délre haladva szokás indexelni, a koordináta-rendszer, ahol a rácshálót elhelyezzük, pedig a legtöbbször északkeleti tájolású.

Az affin transzformáció mátrixok segítségével is felírható a

$$\begin{bmatrix} T_1 & T_2 & X_0 \\ T_3 & T_4 & Y_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} o \\ s \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{2.2.2}$$

alakban. Ha kifejtjük a szorzás eredményeként kapott mátrix X és Y elemeit, akkor megkapjuk a 2.2.1-ben látható összefüggéseket.

Ha egy geodéziai koordinátából akarjuk megmondani, hogy a rácsháló rendszerében hová esik az adott pozíció, akkor az előző transzformációnak az inverzével a

$$\begin{bmatrix} T_1 & T_2 & X_0 \\ T_3 & T_4 & Y_0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \cdot \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} o \\ s \\ 1 \end{bmatrix} \tag{2.2.3}$$

[8] A témával kapcsolatban a http://www.geod.bme.hu/on_line/etrs2eov/etrs2eov_doc.html oldalon találunk részletes leírást. Az geodéziai pontosságú EOV transzformációkhoz a https://github.com/OSGeoLabBp/eov2etrs oldalról lehet olyan javító rácsokat letölteni, amelyekkel az EHT2014 szolgáltatással 1 centiméteren belül megegyező átszámításokat lehet végezni. Ezek az anyagok a BME Általános és Felsőgeodézia Tanszékén készültek, Siki Zoltán és Takács Bence munkái.

összefüggéssel kell dolgozni. A sorok és az oszlopok sorszámára ilyenkor általános esetben nem egész számokat kapunk, a keresett pont magasságát ezért a legközelebbi rácspontok magasságából valamilyen interpolációs eljárással kell meghatározni. (Erről bővebben a 2.4.1 részben lesz szó.)

A rácsháló oszloponkénti illetve soronkénti rácsállandóját a $\tau_o = \sqrt{T_1^2 + T_3^2}$ és a $\tau_s = \sqrt{T_2^2 + T_4^2}$ képletekkel tudjuk kiszámítani. A $T_2 = T_3 = 0$ esetben (a rácsháló vonalai ekkor párhuzamosak a koordinátarendszer tengelyeivel) ezek a $\tau_o = |T_1|$ és $\tau_s = |T_4|$ alakokra egyszerűsödnek.

Még ha nem is párhuzamosak a koordinátarendszer tengelyeivel, a rácsháló vonalai általában merőlegesek egymásra. Ilyenkor a (T_1, T_2) és a (T_3, T_4) vektorok skalárszorzata nulla.

A GRID modellek és raszter adatok georeferálásánál nagyon fontos tisztázni, hogy az alkalmazott program a kétdimenziós adatsor térbeli vonatkozását hogyan értelmezi. Az eddig bemutatott elv az adatokat egy rácsháló rácspontjainak tekintette (a GRID modellek elvének megfelelően), de sok programban tárolt kétdimenziós tömb elemeit egy képelem (raszter) középpontjaként kezelik. A helytelen értelmezés a rácsállandó (felbontás) felének megfelelő nagyságú hibát eredményez.

2.3. TIN modellek

A domborzat digitális ábrázolásának a szabályos rácsháló alkalmazása mellett a másik elterjedt módszere, hogy szabálytalanul elhelyezkedő támpontokra háromszög alakú felületdarabokat illesztünk, melyek összessége fogja meghatározni a domborzat modellezett felületét. Ezt a modellt az angol „Triangulated Irregular Network” (szabálytalan háromszögháló) kifejezés rövidítéseként TIN modellnek nevezzük. [105]

Ennek a modellnek előnye, hogy támpontjait mindig a szükséges helyeken és mindenhol a szükséges sűrűségben tudjuk felvenni, így a lehetőség nyílik jobban lekövetni különféle domborzati formákat. Hátránya, hogy a támpontok rendezetlen halmazának és a közöttük lévő kapcsolatoknak a tárolása jóval bonyolultabb mint a másik modell esetében fellépő hasonló feladatok, és egy pont magasságának a megállapítása is összetettebb feladat. [1]

2.3.1. A TIN modellek meghatározása, létrehozása

Egy sík pontjainak egy tetszőleges halmazához egy háromszöghálót tudunk rendelni a Delaunay háromszögelés módszerével. Az így létrejövő háromszögháló minden háromszögéről elmondható, hogy a köré írt kör nem tartalmazza a ponthalmaz egyetlen pontját sem, a háromszögnek a körön elhelyezkedő három csúcsát leszámítva. Igaz továbbá az az állítás is, hogy két szomszédos háromszög esetén a közös oldallal szemben lévő szögek összege nem lehet nagyobb 180 foknál.

A Delaunay háromszögelés matematikai szempontból is fontos, kapcsolódik sokféle matematikai problémához, szoros összefüggésben van például a Voronoj sokszögeléssel (duális feladatok). A Voronoj sokszögelés megoldásának egyik módja is a Delaunay háromszögelésre való visszavezetés. A Delaunay háromszögeléssel kapott háromszögháló a gyakorlati alkalmazások szempontjából is előnyös tulajdonságokkal rendelkezik, mivel a háromszögek alakja a fent ismertetett összefüggések miatt jobban közelít a szabályos háromszögekhez, mint más megoldások esetén.

A Delaunay háromszögelés megoldására sokféle algoritmus ismert, a [87] két a gyakorlatban is jól használható algoritmust is ajánl. A probléma visszavezethető egy eggyel nagyobb dimenziójú térben történő befoglaló konvex burok számítására, amennyiben a pontokat egy paraboloid felületére vetítjük. A [96] hatékony algoritmusokat mutat be többek között a Delaunay háromszöghálók generálására is.

A probléma matematikai értelemben történő megoldhatóságához a ponthalmaz semelyik három pontja nem eshet egy egyenesre, illetve semelyik négy pontja nem illeszkedhet egy körre. Az egyik legegyszerűbb példa a feltételt nem teljesítő ponthalmazra egy négyzet négy csúcsa. Erre a ponthalmazra kétféleképpen is illeszthetünk két háromszöget, közös oldaluk lehet a négyzet bármelyik átlója, a háromszögek köré írt körök íve viszont mind a négy pontot tartalmazni fogja. Ez nem csupán egy elméleti lehetőség, a gyakorlatban is sokszor előfordul, amikor egy szabályos négyzetrácshálón alapuló GRID modell rácspontjaira illesztünk egy TIN hálót.

Az algoritmusok implementációja során természetesen az ilyen kivételekre is fel kell készülni. Egy matematikai jellegű számításokat végző alkalmazásban ilyenkor a program általában jelzi, hogy a feladnak nincs (valódi) megoldása, mint ahogyan például egy azonosságot tartalmazó egyenletet sem old meg egy számítógépes algebrai rendszer. Térinformatikai (és minden más műszaki jellegű) alkalmazások esetében elvárható viszont, hogy az ilyen esetekre is kapjunk valamilyen kielégítő megoldást, a korábbi négyzetes példa esetében például jöjjön létre háromszögháló a két lehetséges megoldás akármelyikével.

Műszaki célokra a Delaunay háromszögelés olyan módosított változatait szokás alkalmazni, amelyek lehetővé teszik a háromszögháló létrehozásába való beavatkozást is egyes élek direkt megadásával. Erre akkor lehet szükség, ha a felmérés során meghatároztuk a terepnek olyan törésvonalait (pl. rézsűk széle, árkok vonalai), amelyeket a terepfelszín reprezentációjaként létrehozott háromszöghálóban is mindenféleképpen élekként szeretnénk kezelni. A Delaunay háromszögelésnek egy ilyen fajtájára kínál megoldást a [49], illetve a többdimenziós esetre a [119].

A Delaunay háromszögelés háromszögei összességükben a ponthalmaz konvex befoglaló burkát adják. Konkáv alakú munkaterület esetén zavaró lehet, hogy a felületmodellünk olyan részeket is tartalmaz, amelyet valójában nem is mértünk fel. Ezt a háromszögek területeire vagy oldalaik hosszára bevezetett feltétellel, egy a megadott limitnél nagyobb elemek eltávolításával lehet kiküszöbölni; vagy akár a munkaterületet határoló élek megadásával is kezelhető a probléma. Ilyen módon akár szigetszerű belső területek is kihagyhatóak a domborzatmodellből.

Bizonyos esetekben felmerülhet az igény, hogy a háromszögháló háromszögeinek oldalai ne legyenek hosszabbak egy meghatározott értéknél, illetve a mérettől függetlenül ne legyenek a szabályostól túlzottan eltérő háromszögek. A [118, 120] erre a problémára kínál algoritmusokat, amelyek segítségével újabb pontok beiktatásával lehet „finomítani" a háromszöghálót.

További algoritmusokat ismertet a Delaunay háromszögelésre a [67, 124, 121, 51].

A TIN háló kialakításakor csak a vízszintes koordinátákat vesszük figyelembe, de természetesen a létrejövő domborzatmodell a pontokhoz tartozó magassági adatok révén válik teljessé. A Delaunay háromszögelés egyébként értelmezhető magasabb dimenziókban is, de ha sík (2D) helyett térben (3D) végezzük el, akkor háromszögek helyett tetraédereket ad eredményül.

2.3.2. A TIN modellek tárolásának kérdései

A TIN modell háromszöghálója háromféle alapelemből épül fel: lap, él és csúcs. Ezek az elemek kapcsolatban állnak egymással. Az élek két csúcsot kötnek össze és két szomszédos lap határát képezik, vagyis a szomszédos lapok háromszögeinek közös oldalai. (Kivéve a háromszögháló szélén elhelyezkedő éleket, amelyek csak egyetlen lapot határolnak.) Minden lap három darab csúcson fekszik, amelyeket három darab él is összeköt, miként egy háromszög is három ponttal adható meg, és három oldala van. Egy csúcs kettő vagy több élhez, és egy vagy több laphoz kapcsolódik. Mivel a csúcsok elhelyezkedése szabálytalan, az élek és a lapok elhelyezkedése is az.

A TIN modell tárolására használt adathalmazt úgy kell felépíteni, hogy könnyen (vagyis minél kevesebb műveleti lépéssel) elvégezhetőek legyenek azok a lekérdezések, amelyek a modellel kapcsolatos műveletek alapját képezik. Fontos, hogy gyorsan megállapítható legyen, hogy egy vízszintes koordinátáival adott pozíció melyik háromszögbe esik, vagy hogy egy térrészbe mely háromszögek esnek; valamint egyszerűen lekérdezhetőnek kell lennie az elemek közötti topológiai kapcsolatoknak is.

A TIN domborzatmodell által meghatározott felület a számítógépes grafikában használt felülethálóként (mesh) is felfogható, aminek egy olyan speciális esete, ahol valamennyi felületelem háromszög, és ezen háromszögeknek a vízszintes síkra vetített képei között nincs átfedés. A felületháló általános esetben a térben tetszőlegesen elhelyezkedő, egymáshoz közös éleik (edge) mentén kapcsolódó sokszögekből (face) áll. Az élek csúcsokat (vertex) kötnek össze, amelyek a poligonháló sokszög alakú felületelemeit is meghatározzák. Természetesen a TIN modellhez hasonlóan a GRID modellek is leképezhetőek egy felülethálóra, ilyenkor annak elemei nem háromszögek lesznek, hanem olyan négyszögek, amelyek vízszintes vetületei szabályos négyzetek.

A felülethálók tárolására alkalmazni lehet például a „szárnyas-él” modellt [32, 33], ami így a TIN esetében is használható. A szárnyas él modell központi eleme az él, ami tartalmazza a kezdő- és végpontra való hivatkozásokat (mutatók vagy azonosítók, az implementációtól függően), az éltől az előző pontok sorrendje által meghatározott irány szerint értelmezve balra illetve jobbra eső felületelemre való hivatkozásokat, illetve arra a két másik élre való hivatkozást, amelyekkel az ennek két felületnek a körbejárását tudjuk folytatni az él végpontját követően. A felületelemek és a csúcsok tárolásához az egyik kapcsolódó élre való hivatkozást kell tárolni mindkét esetben, illetve a csúcsok esetében természetesen a koordinátákat is.

Az, hogy bármilyen TIN vagy GRID modellből egyszerűen levezethető egy mesh azért fontos, mert azt utána sokféle számítógépes grafikai alkalmazásban használhatjuk a domborzat felszínének megjelenítésére. Ha textúrázott megjelenítést szeretnénk, akkor a vízszintes koordinátákat könnyen használhatjuk például egy digitális ortofotónak a felszínre illesztéséhez, mert a vízszintes koordinátákból egyszerűen át tudunk térni a textúra koordináta-rendszerébe.

Relációs adatbázisban tárolva a TIN modell háromszöghálóját a szárnyas él modellhez képest elhagyhatóvá vállnak a lapok (hacsak nem akarunk hozzájuk kapcsolódva valamilyen adatot tárolni) és az éleknél a továbbhaladó élekre való hivatkozások, annak köszönhetően, hogy az élek adatait tároló táblát többféle tulajdonság szerint is indexelni tudjuk.

A GML[9] adatokban a `TinType` típusú objektumok használatával lehet a szabálytalan háromszöghálókon alapuló felületmodelleket kezelni. Ez a típus pontjaival, törésvonalaival és határvonalaival tárolja a TIN modellt; az ezekből levezethető háromszögháló számítása a feldolgozó program feladata. Egy csak háromszögekből álló, de egyébként tetszőleges felülethálót a `TriangulatedSurface` típusú, egy a korábban bemutatott „mesh"-nek megfelelő tetszőleges sokszögekből összeállított felülethálót pedig a `PolyhedralSurface` típusú objektumokkal lehet modellezni.

Az OGC[10] Simple feature access[11] szabványsorozatában a `PolyhedralSurface` objektumtípus alkalmas TIN modellek tárolására. Az OGC 06-103r4 szabvány 15. számú ábrája egy olyan UML class diagramot tartalmaz, amin egy a `PolyhedralSurface` osztályból öröklődő `TIN` osztály látható, de az OGC 06-104r4 szabvány ilyen típust nem ismer.

2.4. A domborzat egy pontjának jellemzése

Az alábbiakban a domborzat olyan tulajdonságait foglalom össze, amelyekkel a domborzat egyes pontjai jellemezhetőek. Ezt úgy is mondhatjuk, hogy pontonként változó tulajdonságokról van szó.

2.4.1. Pontonként számítható számszerű jellemzők

Amennyiben a domborzat egy pontját akarjuk jellemezni, a levezethető jellemzők megadásához a legegyszerűbb a domborzatot egy kétváltozós függvénynek tekinteni, így azt matematikai eszközökkel tudjuk a továbbiakban kezelni. Ha a domborzat egy kétváltozós függvény, akkor a terepfelszín magasságát egy pontban a következő módon írhatjuk fel:

$$h = f(x, y) \tag{2.4.1}$$

Ennek a felületnek kiszámíthatjuk az x illetve y irányú deriváltjait, amelyeket többféleképpen is szokás jelölni (én a továbbiakban a p és q jelölést fogom használni) és együttesen a gradiensvektort adják:

[9] A GML (Geographic Markup Language) szabvány elérhető a http://www.opengeospatial.org/standards/gml oldalon. A GML-t az ISO is szabványosította ISO 19136:2007 számon.

[10] Open Geospatial Consortium, honlapja a http://www.opengeospatial.org/ címen érhető el.

[11] A szabványsorozatban használt közös ismereteket az OGC 06-103r4 (elérhető a http://www.opengeospatial.org/standards/sfa címen) tartalmazza. A gyakorlatban a legtöbbet a szabványsorozatnak az OGC 06-104r4 jelű tagját (elérhető a http://www.opengeospatial.org/standards/sfs címen) alkalmazzák, ami az SQL alapú adatbázis-kezelők térinformatikai funkciókkal való kiterjesztésére határoz meg egy egységes megoldást.

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} f'_x \\ f'_y \end{bmatrix} = \begin{bmatrix} p \\ q \end{bmatrix} \tag{2.4.2}$$

Az esésvonal iránya ellentétes lesz a gradiensvektor irányával. Irányszögére igaz a következő összefüggés:

$$\delta_{evon} = \arctan\left(\frac{-q}{-p}\right) \tag{2.4.3}$$

A gyakorlatban ez a $-p$ és $-q$ koordinátakülönbségekhez tartozó irányszög számítását jelenti. Ez legtöbb programozási nyelvben az `atan2` függvény segítségével oldható meg a legegyszerűbben, ami a $-p$ és $-q$ értékek előjeleiből kikövetkezteti a helyes szögnegyedet és a $p = 0$ esetet is megfelelően kezeli. A csapásvonal iránya az esésvonal (illetve a gradiensvektor) irányára merőleges:

$$\delta_0 = \delta_{evon} \pm 90^\circ \tag{2.4.4}$$

A legnagyobb terepesést a gradiensvektor hossza adja meg:

$$g_{evon} = |\nabla f| = \sqrt{p^2 + q^2} \tag{2.4.5}$$

Egy a ponton átmenő tetszőleges irányhoz (jelölése: δ) tartozó terepesést is ki lehet számítani:

$$g_\delta = g_{evon} \cdot \cos\left(\delta - \delta_{evon}\right) = -p \cdot \cos\delta - q \cdot \sin\delta \tag{2.4.6}$$

Számítani lehet egy megadott terepeséshez (jelölése: g) tartozó irányszöget is, feltéve hogy $|g| \leqq g_{evon}$, vagyis ez a terepesés nem nagyobb legnagyobb terepesésnél:

$$\delta_g = \arccos\frac{p \cdot g \pm \sqrt{g_{evon}^2 - g^2}}{g_{evon}^2} = \arccos\frac{p \cdot g \pm \sqrt{p^2 + q^2 - g^2}}{p^2 + q^2} \tag{2.4.7}$$

Az arkusz koszinusz függvénynek több lehetséges értéke is lehet, ezek mindegyikéhez egy-egy olyan irányszög tartozik, amelyben a terepesés értéke a keresett érték. A fenti képletekben g betűvel, illetve annak különféle alsó indexes változataival jelölt mennyiségek helyett a terepviszonyok kifejezésére használhatjuk a lejtőszöget is. A kétféle mennyiség között a következő összefüggések segítségével nyílik lehetőségünk az átszámításra:

$$\alpha = \arctan g \tag{2.4.8}$$

$$g = \tan\alpha \tag{2.4.9}$$

A g-vel jelölt mennyiségeket sokszor százalékos formában adják meg. Például a $g = 0.15$ helyett azt is mondhatjuk, hogy a lejtő 15 százalékos, illetve a 2.4.8 segítségével azt is kiszámíthatjuk, hogy a lejtőszög $\alpha \simeq 8.53^\circ$.

A felület görbületi viszonyainak [93] leírásához már a második deriváltakra is szükségünk van. Ezeket általában az ún. Hesse-mátrixban szokás megadni:

$$\nabla^2 f = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} = \begin{bmatrix} r & s \\ s & t \end{bmatrix} \tag{2.4.10}$$

A továbbiakban a fent is használt r, s és t jelöléseket fogom használni ezekre a mennyiségekre. A felület legnagyobb és legkisebb görbülete a következő képletekkel számítható:

$$\lambda_{1,2} = \frac{r + t \pm \sqrt{(r+t)^2 + 4(rt - s^2)}}{2} \tag{2.4.11}$$

A legnagyobb görbület irányát a következő módon kapjuk meg:

$$\delta_{cmax} = \frac{1}{2} \cdot \arctan \frac{2s}{r - t} \tag{2.4.12}$$

A legkisebb görbület iránya erre merőleges. Az irányszög számításánál itt is érvényesek a gradiensvektor irányánál leírtak.

A domborzat egyszerű jellemzőit számítani tudjuk a fenti képletekben h, p, q, r, t és s betűkkel jelzett jellemzők ismeretében, vagyis ha meg tudjuk állapítani a domborzatnak mint kétváltozós függvénynek a helyettesítési értékét (a magasságot) valamint az első és második differenciálhányadosait. Akár síkokat is használhatunk a domborzatmodell alapelemeinek, a helyettesítési érték (h) és az első differenciálhányadosok (p és q) értékeire használható számokat kapunk, a magasságot és a lejtésviszonyokat ebből elfogadhatóan ki lehet számítani.

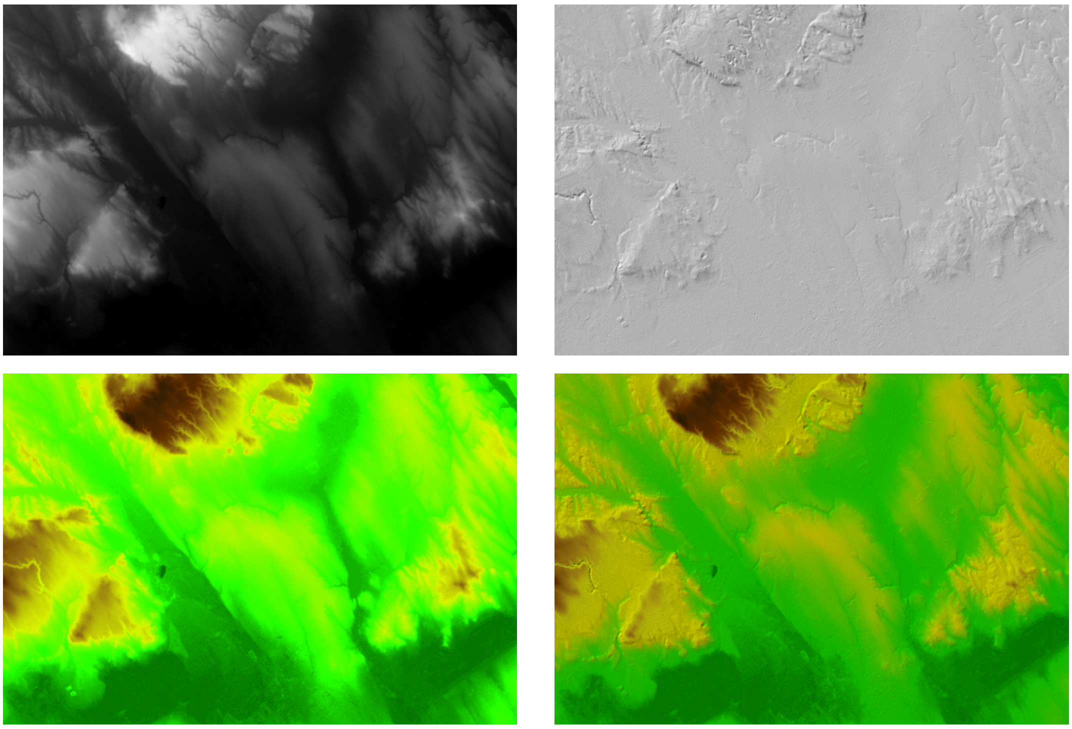
Amennyiben a terepfelszín görbületét vagy egyéb a második differenciálhányadosoktól (r, t és s) is függő jellemzőket akarunk tanulmányozni, már semmiféleképpen sem megfelelő a síkok alkalmazása, mert a sík minden pontjában $r = t = s = 0$, így a görbületre vonatkozó információhoz ebből nem juthatunk. (De akár matematikai összefüggések nélkül is könnyen belátható, hogy síkokkal nem lehet görbületi viszonyokat modellezni.) A bilineáris felület sem alkalmas a görbületi viszonyok tanulmányozására, mert két irányban is mindig egyenes a keresztmetszete. Ezeknek a jellemzőknek a meghatározásához ilyen esetekben magasabb fokú felületeket kell alkalmazni.

2.4.2. Pontonként számítható jellemzők grafikus ábrázolása

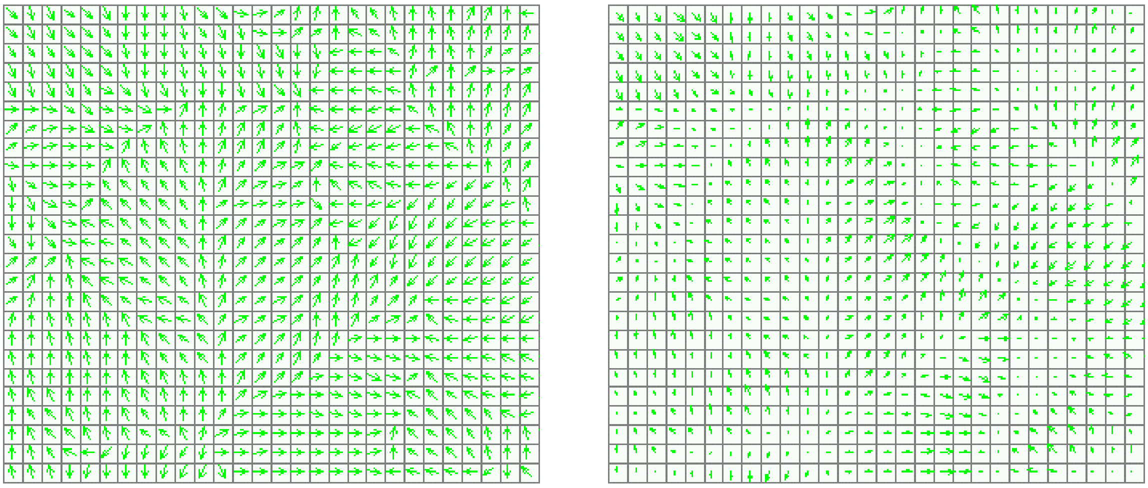
Felmerül az igény, hogy a terület minden pontjában kiszámított értékeket egy tematikus térképen ábrázoljuk. Ez történhet úgy, hogy meghatározott értékekhez színeket rendelünk, majd ezekkel a színnel és a közöttük található átmeneti árnyalatokkal ábrázoljuk az előforduló értékeket. Ennek a legismertebb példája az, amikor a magasságot ábrázoljuk színfokozatokkal, a zöldtől indulva a sárgán és a banán át a feketéig.

Más számszerű jellemzők is ábrázolhatóak ilyen módon. Például a (legnagyobb) terepesés. Különféle színárnyalatokkal tudjuk kifejezni, hogy a terep felszíne az egyes pontokban milyen szöget zár be a vízszintessel, a meredekebb részeket így másmilyen színárnyalatok fogják megjeleníteni, mint a közel sík területeket.

Érdekes a helyzet az irányszög jellegű mennyiségeknél, például az esésvonal irányánál. Itt is lehetne alkalmazni az előbbi módszert, de az a kezdő irány környékén érdekes



2.4.1. ábra. A domborzat ábrázolása különféle módszerekkel: a magasságok ábrázolása szürke árnyalatokkal (bal felső kép), a magasságok ábrázolása a földrajzi térképeken szokásos színskála segítségével (bal alsó kép), a domborzat ábrázolása árnyalásos módszerrel (jobb felső kép) valamint a színskála és az árnyalásos ábrázolás kombinációja (jobb alsó kép). A képek 25 méteres újramintavételezéssel EOV rendszerbe transzformált 1 másodperces felbontású SRTM modell alapján készültek QGIS programmal.



2.4.2. ábra. A lejtő irányának megjelenítése nyilakkal. A bal oldali képen minden nyíl mérete egyforma. A jobb oldali képen a nyilak hossza a terep lejtésével arányos. A képek a GRASS[101] felhasználásával készültek.

következményekkel járna, mivel a skála két végpontja itt egymás mellé kerülne. Hogy ne legyen itt az alig eltérő irányokhoz rendelt színárnyalatok között törésszerű eltérés, célszerű a színskála két végpontjának ugyanazt a színt megadni. Ez történhet a szürke különféle árnyalataival úgy, hogy az egyes irányokhoz azokat a színárnyalatokat rendeljük, amelyek egy oldalról megvilágított kúp azonos irányú alkotójához tartoznának. Ezzel a módszerrel egy látványos, az árnyalásos domborzatábrázoláshoz hasonló képhez jutunk, aminek hátránya, hogy a megjelenített kép egy színéhez több irány is tartozik. A problémára megoldást jelenthet, ha az egyes irányokhoz egy színkörnek a megfelelő színeit rendeljük, amivel elérhető, hogy minden irányhoz eltérő színárnyalatok tartozzanak.

A lejtő irányát jelölhetjük az adott irányba mutató nyilakkal. A nyilak mérete lehet egységnyi, vagy arányos lehet a terepeséssel. (2.4.2 ábra) Az így készült térképeket gradienstérképnek nevezzük. Hátránya, hogy a nyilak kirajzolása jóval több helyet igényel mint egy képpont kiszínezése, viszont az így készült kép szemléletes.

2.4.3. Pontonként számítható tulajdonságokból származtatható jellemzők

A gyakorlatban a terep lejtésviszonyainak a jellemzéséhez nem közvetlenül a gradiensvektort és a belőle származtatható egyéb geometriai adatokat használják, hanem az esésvonal irányából és a terepesés értékéből kategóriába sorolással képeznek jellemzőket, a lejtőkategóriát és a kitettséget. A lejtőkategória meghatározásához az 2.4.1 táblázat szerint sorolják be a területeket a százalékban kifejezett lejtés alapján névvel és (római) számmal is jelzett kategóriákba.

A lejtőkategóriákat a mezőgazdasági művelés szempontjai szerint állapították meg. Azért tekintjük az 5% alatti lejtésű területeket síknak, mert a tapasztalatok szerint ezeken a helyeken az erózió még nem jelentős. Az V. kategóriába tartozó (meredek) területek mezőgazdasági művelésre már nem alkalmasak. A 2.4.2 táblázatban a lejtők kategorizá-

2.4.1. táblázat. A lejtőkategóriák a lejtés függvényében

lejtőkategória	lejtés	minősítés
I.	5%-ig	sík
II.	5% és 12% között	enyhén lejtős
III.	12% és 17% között	lejtős
IV.	17% és 25% között	enyhén meredek
V.	25% felett	meredek

2.4.2. táblázat. A lejtők kategorizálásának egy másik módja. ([63] alapján)

típus	lejtőszög	megjegyzés
enyhe lejtő	15°-ig	gyalogosan és gépjárművel is könnyen járható
meredek lejtő	15° és 30° között	gyalogosan és terepjáróval járható
igen meredek lejtő	30° és 45° között	gyalogosan is nehezen járható
falszerű lejtő	45° felett	csak hegymászó felszereléssel járható

lásának egy másik, a terep járhatóságát szem előtt tartó módját láthatjuk.

A terep lejtésének iránya alapján is kategóriákba sorolhatjuk egy terület pontjait. A nem sík területek kitettségének megállapítása az esésvonal irányszöge alapján történik, a 2.4.3 táblázat szerint.

Ez a beosztás szintén a növényzet igényeihez és a mezőgazdasági hasznosíthatósághoz kapcsolódik, és ebben a formában a Föld északi féltekére érvényes. A kategóriák közötti választóvonalak nem a $45° + k \cdot 90°$-os irányokban vannak, hanem ahhoz képest 1/16-od körívnyivel (vagyis 22,5°-al) elforgatva, a $67,5° + k \cdot 90°$-os irányokban.

2.5. Interpolációs módszerek

Ebben a részben a 2.4.1 részben bemutatott jellemzők (a terep magassága, valamint a kétváltozós függvénynek tekintett terepfelszín első és második deriváltjai) kiszámításával foglalkozom különféle modellekre illesztett különféle felületek segítségével

2.5.1. Interpoláció síkkal

A TIN háló egy háromszögére kézenfekvő módon lehet síkot illeszteni, mivel a háromszög három csúcsa egyértelműen meghatároz egy síkot. Amennyiben a keresett, vízszintes koordinátáival megadott pozíció a háromszög területére (a csúcsok vízszintes koordinátái által meghatározott háromszög belsejébe) esik, a terep magasságát a pozícióhoz

2.4.3. táblázat. A kitettségek kategóriái az esésvonal irányszögének függvényében

az esésvonal irányszöge	égtáj	kitettség
0° és 67° valamint 338° és 360° között	É, ÉK	III.
68° és 157° között	K, DK	II.
158° és 247° között	D, DNY	I.
248° és 337° között	NY, ÉNY	II.

ezen a síkon tartozó magasságként kell értelmezni.

Ha a háromszög csúcsainak koordinátáit x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3 és z_3 betűkkel jelöljük, az x és y tetszőleges vízszintes koordinátákkal megadott pozícióban az erre a három pontra illesztett sík magasságát pedig z-vel, akkor a sík egyenletét azt az összefüggést felhasználva írhatjuk fel, hogy a négy pont által alkotott tetraéder térfogata nulla. A tetraéder térfogatát determinánsos alakban felírva az egyenlet a következőképpen fog kinézni:

$$\frac{1}{6}\begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix} = 0 \tag{2.5.1}$$

A fenti egyenletet hattal szorozva és a determinánst az első sora szerint kifejtve kapjuk:

$$x\begin{vmatrix} y_1 & z_1 & 1 \\ y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \end{vmatrix} - y\begin{vmatrix} x_1 & z_1 & 1 \\ x_2 & z_2 & 1 \\ x_3 & z_3 & 1 \end{vmatrix} + z\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} - \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix} = 0 \tag{2.5.2}$$

Amit z-re rendezve kapjuk:

$$z = \frac{\begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}} - \frac{\begin{vmatrix} y_1 & z_1 & 1 \\ y_2 & z_2 & 1 \\ y_3 & z_3 & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}}x + \frac{\begin{vmatrix} x_1 & z_1 & 1 \\ x_2 & z_2 & 1 \\ x_3 & z_3 & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}}y \tag{2.5.3}$$

A fenti összefüggésben sík egyenletének három paramétere determinánsok hányadosaként áll elő, ahol a nevező mindhárom esetben azonos. A determinánsokat kifejtve kaphatjuk a következő értékeket:

$$\begin{gathered} D = x_1y_2 + x_3y_1 + x_2y_3 - x_1y_3 - x_2y_1 - x_3y_2 \\ p = \frac{y_1z_3 - y_2z_1 - y_3z_2 - y_1z_2 - y_3z_1 - y_2z_3}{D} \\ q = \frac{x_1z_2 + x_3z_1 + x_2z_3 - x_1z_3 - x_2z_1 - x_3z_2}{D} \\ M_0 = \frac{x_1y_2z_3 + x_3y_1z_2 + x_2y_3z_1 - x_3y_2z_1 - x_2y_1z_3 - x_1y_3z_2}{D} \end{gathered} \tag{2.5.4}$$

Ezekkel a paraméterekkel már egyszerűbben felírható a sík egyenlete:

$$h = M_0 + px + qy \tag{2.5.5}$$

A terepfelszínnek az x, y pozícióban vett magasságát itt már ismét h-val jelöltem a 2.4.1 jelölésének megfelelően. A p és a q mennyiségek a 2.4.2 egyenletben hasonló módon jelölt mennyiségekkel egyeznek meg, vagyis a gradiensvektor elemei. Az M_0a síknak a koordináta-rendszer kezdőpontjában vett magassága. A D csak egy segédmennyiség, amit azért vezettünk be, hogy a 2.5.3 nevezőiben található determináns értékét csak egyszer kelljen kiszámítani. A D értéke akkor nulla, ha a három pont vízszintes vetülete egy egyenesre esik. Ilyenkor a számításokat nem lehet elvégezni, de azoknak geometriai értelme sem lenne.

Ha a három pont által meghatározott sík merőleges (normális) irányát akarjuk meghatározni, akkor arra egyszerűbb módszer is kínálkozik. Ha vesszük bármelyik pontból a másik kettő pontba mutató vektorok vektoriális szorzatát, annak iránya merőleges lesz a három pont közös síkjára.

GRID háló esetében a négyzetet egy átlója mentén két háromszögre lehet bontani, amelyekre az előzőekben bemutatott módon lehet síkot illeszteni, vagy a háló egy négyzetén belül a négy sarokpontra illeszkedő bilineáris felülettel lehet dolgozni a következőekben bemutatott módon.

2.5.2. Bilineáris felület alkalmazása

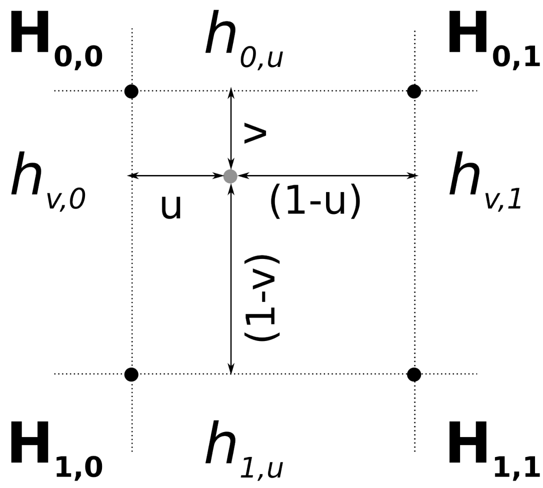
A bilineáris felület illesztése során a vizsgált pontot tartalmazó négyzet négy pontjából indulunk ki. Vesszük a négyzet két párhuzamos oldalát, majd lineáris összefüggésekkel számítjuk annak a pontnak a magasságát, amelyek a vizsgált pont merőleges vetületeiben helyezkednek el, vagyis a szakaszok változó koordinátái a vizsgált pont kérdéses koordinátájával egyeznek meg. A következő lépésben az így kapott két pont között egy hasonló lineáris interpolációval számíthatjuk ki a vizsgált pont magasságát. Mindegy, hogy a négyzet melyik két párhuzamos oldalával kezdünk, ugyanazt az eredményt kapjuk.

Egy szakaszon belül egy a szakaszt két kisebb szakaszra osztó pont (pozíció) magasságát (vagy bármelyik más koordinátáját) úgy lehet számítani, hogy a végpontok magasságainak vesszük a velük szemben lévő részszakaszok hosszaival súlyozott számtani közepét. Egy egységnyi oldalhosszúságú négyzettel dolgozva (vagy egy egyszerű osztással ilyenre áttérve) a 0 és 1 közötti értékeket felvehető u-val és v-vel jelölhetjük a vizsgált pozíció helyzetét, a négy rácspont magasságát pedig a $H_{0,0}$, $H_{0,1}$, $H_{1,0}$ és $H_{1,1}$ jelölésekkel (2.5.1 ábra). A vizsgált pozíció magassága így:

$$H = H_{0,0}\left(1-u\right)\left(1-v\right) + H_{0,1}u\left(1-v\right) + H_{1,0}\left(1-u\right)v + H_{1,1}uv \tag{2.5.6}$$

Ha a vizsgált pozíción keresztül húzott, a rácsháló vonalaival párhuzamos két vonallal a rácsháló négyzetét négy részre bontjuk, akkor az egyes rácspontok magasságainak a velük szemben elhelyezkedő téglalap területével súlyozott számtani közepe lesz a bilineáris felület magassága a vizsgált pozícióban.

A számítások során $H_{0,0}$-al jelölt rácspont elhelyezkedését (indexeit) a teljes rácson belül a 2.2.3 képlet segítségével számítható o és s értékek egész részeként kaphatjuk meg. Az u és a v paramétereket az előbbi értékek eggyel vett osztási maradékaként állíthatjuk elő.



2.5.1. ábra. A vizsgált pozíció magasságának meghatározása a rácsháló környező pontjainak magasságaiból kiindulva, bilineáris interpoláció segítségével.

2.5.3. Interpoláció polinomokkal

Ha GRID modell egy négyzete fölé magasabb fokú polinomok segítségével akarunk egy felületet meghatározni, akkor a négyzet sarokpontjain túl néhány további szomszédos pont bevonására is szükségünk van. Az bilineáris felületeknél 0 és 1 indexekkel jelölt rácspontok mellett a szomszédos -1 és 2 sorszámmal jelölhető indexű, vagy akár további elemek értékeit is fel kell használni a számítások során.

A terepfelszín magasságát egy magasabb fokú polinomot használva a

$$h\left(x,y\right)=\sum a_{i,j}x^{i}y^{j} \tag{2.5.7}$$

képlettel adhatjuk meg, ahol az $a_{i,j}$ értékek a polinom paraméterei, amelyekben az i és a j értékei olyan nem negatív egész számok amelyekre az $i+j\leq k$ feltétel teljesül, ahol a k a polinom fokszáma. A $k=1$ eset a sík, a másodfokú illetve harmadfokú felületeket a $k=2$ illetve $k=3$ értékekkel kapjuk. Az előbbi feltétel helyett az $i\leq k'$ és a $j\leq k'$ feltételek együttes teljesülését is elő lehet írni. A bilineáris felület ennek a $k'=1$ esete.

A polinom $a_{i,j}$-vel jelölt paramétereinek meghatározásakor a 2.5.7 összefüggésben a magasságot (és természetesen az x-el és y-al jelölt vízszintes koordinátákat is) ismertnek véve egy a polinom paramétereit ismeretlenként tartalmazó lineáris egyenletet kapunk. A paraméterek számának megfelelő ilyen egyenletet felírva (amihez nyilvánvalóan ilyen számú ismert magasságú pontra van szükségünk) egy olyan lineáris egyenletrendszerhez jutunk, aminek a megoldásával a polinom paramétereit kapjuk. A szükségesnél több egyenlet (több pont) használatakor kiegyenlítéssel határozhatjuk meg a polinom paramétereit.

Az i és j paraméterekre megállapított feltétel fajtájától függ az $a_{i,j}$-vel jelölt paraméterek száma. Erre az első esetben a $\frac{k(k-1)}{2}$ jön ki, vagyis háromszögszámokat kapunk, míg a második esetben a paraméterek száma k'^{2}, vagyis négyzetszámok lesznek. A négyzetszámok előnyösebbek, amikor egy azonos számú sorból és oszlopból álló rácshálóra kell egy felületet illeszteni, mert ilyenkor az adatok száma és a belőlük levezetendő paraméterek száma megegyezik. Hátránya ennek a megoldásnak az, hogy a magasabb fokú tagok nem teljesek: a bilineáris felület egyenletében szerepel például az xy másodfokú szorzat, de x^2 és y^2 tagokat már nem tartalmaz.

A deriválás alapvető szabályait felhasználva a 2.5.7 egyenletből kiindulva számíthatjuk a terepfelszín elsőfokú deriváltjait:

$$\begin{aligned} p &= \tfrac{\partial h}{\partial x} = \sum i a_{i,j} x^{i-1} y^j \\ q &= \tfrac{\partial h}{\partial y} = \sum j a_{i,j} x^i y^{j-1} \end{aligned} \tag{2.5.8}$$

majd ezeket tovább deriválva kapjuk a terepfelszín másodfokú deriváltjainak értékeit megadó összefüggéseket:

$$\begin{aligned} r &= \tfrac{\partial^2 h}{\partial x^2} = \sum i\,(i-1)\,a_{i,j} x^{i-2} y^j \\ t &= \tfrac{\partial^2 h}{\partial y^2} = \sum j\,(j-1)\,a_{i,j} x^i y^{j-2} \\ s &= \tfrac{\partial^2 h}{\partial x \partial y} = \sum i j a_{i,j} x^{i-1} y^{j-1} \end{aligned} \tag{2.5.9}$$

Amikor a terepfelszín egy darabját egy a fentiekben bemutatott polinommal közelítjük, általában harmadfokú polinomot használunk. Az így létrehozott felületek a harmadfokú tagoknak köszönhetően már inflexiót is tartalmazhatnak, de a magasabb fokú polinomoknál kevésbé hajlamosak arra, hogy egy pont megváltoztatásának hatására a felület jelentősen megváltozzon a ponttól távolabbi területeken. A 2.5.7, 2.5.8 és 2.5.9 egyenletek a $k = 3$ (harmadfokú) esetben a következőképpen néznek ki, ha a képletekben szereplő szummázásokat kifejtjük a magasságra:

$$h = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{2,0}x^2 + a_{1,1}xy + a_{0,2}y^2 + a_{3,0}x^3 + a_{2,1}x^2y + a_{1,2}xy^2 + a_{0,3}y^3 \tag{2.5.10}$$

az elsőfokú deriváltakra:

$$\begin{aligned} p &= a_{1,0} + 2a_{2,0}x + a_{1,1}y + 3a_{3,0}x^2 + 2a_{2,1}xy + a_{1,2}y^2 \\ q &= a_{0,1} + a_{1,1}x + 2a_{0,2}y + a_{2,1}x^2 + 2a_{1,2}xy + 3a_{0,3}y^2 \end{aligned} \tag{2.5.11}$$

illetve a másodfokú deriváltakra:

$$\begin{aligned} r &= 2a_{2,0} + 6a_{3,0}x + 2a_{2,1}y \\ t &= 2a_{0,2} + 2a_{1,2}x + 6a_{0,3}y \\ s &= a_{1,1} + 2a_{2,1}x + 2a_{1,2}y \end{aligned} \tag{2.5.12}$$

Gyakorlati okokból, a számítások közben keletkező kerekítési hibák minimalizálása érdekében, az előbbiekben bemutatott, polinomokkal kapcsolatos számításokat célszerű egy olyan eltolt koordináta-rendszerben elvégezni, amelynek az origója a munkaterületre esik.

TIN modellek esetén [62] megad egy módszert, aminek a segítségével a modell háromszög elemeire harmadfokú polinomokat határozhatunk meg. Az eljárás az egyes háromszögekre illesztett polinom felületek között az egymással szomszédos háromszögek esetében folyamatosságot biztosít, a végeselemes számításoknál használthoz hasonló módszert alkalmazva. A vizsgált vízszintes pozícióhoz tartozó háromszöglaphoz rendelt polinomnak az együtthatóiból a 2.5.10 felhasználásával meghatározható a magasság illetve a 2.5.8 és 2.5.9 képletek segítségével az első- és másodfokú deriváltak is.

2.5.4. B-spline felületek alkalmazása

Egy domborzatmodell felbontásának növelése során akár a mesh felületek simítására kidolgozott módszereket is alkalmazhatjuk. Az egyik leginkább elterjedt ilyen módszer a B-spline felületeken alapuló Catmull-Clark-féle felosztás [48] nagyon jól használható olyankor, amikor egy GRID modell felbontását 2^n-szeresére akarjuk növelni.

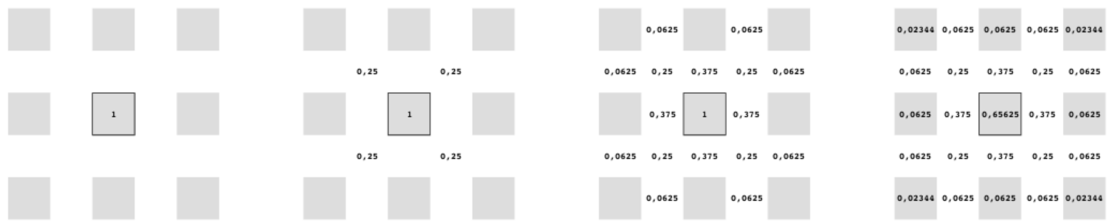
A módszer elve szerint először a négyszögek közepére kell kiszámítani a finomított háló egy új pontját a négy csúcs koordinátáinak átlagaként; majd az élek közelébe eső új pontok következnek, melyek helyzetét a két szomszédos eredeti csúcs és a kettő lapok belsejébe eső, az előző lépés során számított pont koordinátáinak átlagolásával határozzuk meg. A négyszögekből álló felületháló simítása ezzel még nem ér véget, mert az utolsó lépésben az eredeti csúcsoknak is új pozíciót számítunk úgy, hogy az eredeti csúcs koordinátáit $^1/_2$ az előző lépésekben számított nyolc új csúcs (négy a lapok, négy pedig az élek közepének a környékére esik) koordinátáit pedig $^1/_{16}$ súlyokkal átlagoljuk.

Ha a fenti módszert egy GRID domborzatmodell szabályos négyzetrácshálóján alkalmazzuk, akkor a GRID hálójának speciális tulajdonságai következtében az elvégzendő számítások egyszerűsödnek. Az új rácspontok vízszintes értelemben a kiinduló rácspontoktól pontosan egyenlő távolságban fognak elhelyezkedni, a kiinduló rácspontok vízszintes helyzete a finomítás során nem fog megváltozni, vagyis a szokásos számításokat a három térbeli koordináta helyett csak a magasságokkal kell elvégeznünk. A szabályos négyzetrácshálón a 2.5.2 ábrának megfelelően előre ki lehet számolni, hogy egy rácspont magasságát milyen súllyal kell figyelembe venni a finomított rácsháló egyes pontjainak a magasságának a számításakor. A finomított GRID modell rácspontjainak magasságait az 2.5.2 ábra utolsó képen elhelyezkedő számokat megfelelően elrendezve konvolúció-szerűen tudjuk számítani.

A fentiek alkalmazásával egy lépésben kétszeres felbontású (fele akkora rácsközű) modellhez jutunk. A módszert többször egymás után alkalmazva lehetőségünk van 2^n-szeres felbontásúra finomítani a domborzatmodellünket. A súlyokat ehhez akár előre is ki lehet számítani a 2.5.3 ábrán látható módon.

A domborzat felületének a Catmull-Clark-féle módszerrel történő simításakor figyelembe kell venni, hogy ez az eljárás, más simítási módszerekhez hasonlóan, az eredeti pontok koordinátáit (esetünkben csak a magasságokat) is megváltoztatja. Ez egy olyan modellnél, ahol a rácspontok magassága a terep pontos magasságát jelenti nem megengedhető. Ilyen esetekben olyan interpolációs módszereket kell használni, amelyek az eredeti támpontok magasságát változatlanul hagyják, mint például a korábban bemutatott polinom alapú eljárások.

Tapasztalataim szerint a szintvonalas térkép digitális domborzatmodell alapján történő előállításakor látványosan jobb eredményt lehet elérni, ha előtte a domborzatmodell



2.5.2. ábra. A Catmull-Clark-féle módszer alkalmazása egy GRID domborzatmodell négyzetrácshálójának pontjain lépésenként szemléltetve. A szürke hátterű cellák az eredeti rácsháló pontjait jelképezik, a finomított rácsháló a többi (fehér hátterű) cellához tartozóan is rendelkezik rácspontokkal. Az utolsó (jobb szélső) ábrán láthatóak a végleges értékek, amivel a középső (bekeretezett határvonalú) cellához tartozó magasság a finomított rácsháló magasságaihoz hozzájárul.

0,000549	0,001465	0,002930	0,005371	0,005920	0,007813	0,006836	0,007813	0,005920	0,005371	0,002930	0,001465	0,000549
0,001465	0,005859	0,012695	0,021484	0,028809	0,031250	0,031250	0,031250	0,028809	0,021484	0,012695	0,005859	0,001465
0,002930	0,012695	0,029053	0,051758	0,072266	0,085938	0,083984	0,085938	0,072266	0,051758	0,029053	0,012695	0,002930
0,005371	0,021484	0,051758	0,099609	0,149902	0,187500	0,203125	0,187500	0,149902	0,099609	0,051758	0,021484	0,005371
0,005920	0,028809	0,072266	0,149902	0,237610	0,306641	0,342773	0,306641	0,237610	0,149902	0,072266	0,028809	0,005920
0,007813	0,031250	0,085938	0,187500	0,306641	0,414063	0,464844	0,414063	0,306641	0,187500	0,085938	0,031250	0,007813
0,006836	0,031250	0,083984	0,203125	0,342773	0,464844	0,547852	0,464844	0,342773	0,203125	0,083984	0,031250	0,006836
0,007813	0,031250	0,085938	0,187500	0,306641	0,414063	0,464844	0,414063	0,306641	0,187500	0,085938	0,031250	0,007813
0,005920	0,028809	0,072266	0,149902	0,237610	0,306641	0,342773	0,306641	0,237610	0,149902	0,072266	0,028809	0,005920
0,005371	0,021484	0,051758	0,099609	0,149902	0,187500	0,203125	0,187500	0,149902	0,099609	0,051758	0,021484	0,005371
0,002930	0,012695	0,029053	0,051758	0,072266	0,085938	0,083984	0,085938	0,072266	0,051758	0,029053	0,012695	0,002930
0,001465	0,005859	0,012695	0,021484	0,028809	0,031250	0,031250	0,031250	0,028809	0,021484	0,012695	0,005859	0,001465

0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000	0,000000
0,000000	0,000000	0,000013	0,000034	0,000069	0,000126	0,000173	0,000275	0,000343
0,000000	0,000000	0,000034	0,000137	0,000298	0,000504	0,000767	0,001099	0,001526
0,000000	0,000000	0,000069	0,000298	0,000681	0,001213	0,001877	0,002808	0,003784
0,000000	0,000000	0,000126	0,000504	0,001213	0,002335	0,003849	0,005737	0,007996
0,000000	0,000000	0,000173	0,000767	0,001877	0,003849	0,006446	0,009842	0,013710
0,000000	0,000000	0,000275	0,001099	0,002808	0,005737	0,009842	0,015076	0,021393
0,000000	0,000000	0,000343	0,001526	0,003784	0,007996	0,013710	0,021393	0,030602

2.5.3. ábra. A Catmull-Clark-féle módszer többszöri alkalmazása egy GRID domborzatmodell négyzetrácshálójának pontjain a négyszeres (fenti kép) illetve nyolcszoros (lenti kép) felbontás elérése érdekében. Az egyre ritkábban elhelyezkedő szürke hátterű cellák itt is az kiindulási rácsháló pontjait jelképezik. A nyolcszoros felbontást bemutató lenti ábra terjedelmi okokból a számok elhelyezkedésének szimmetriáit kihasználva már csak a terület negyedét mutatja be.

felbontását az előzőekben bemutatott módszerrel növeljük, amint az a 2.5.4 ábrán is látható. Ebben az esetben még az eredeti rácspontok magasságának megváltozása sem jelent gondot, hiszen egy digitális domborzatmodellből előállított szintvonalrajzot inkább csak megjelenítési célokra használunk, azon méréseket nem szokás végezni. [17] A témát a [132] is vizsgálta.

2.6. Geometriai jellegű számítások

Az alábbiakban azokat a legfontosabb elemzéseket mutatom be, ahol a domborzattal mint térbeli felülettel végzünk különféle műveleteket.

2.6.1. Metszet jellegű vonalak létrehozása

Sokféle a felületmodellekkel kapcsolatos feladat vezethető vissza a felületeknek valamilyen másik felülettel képzett metszésvonalainak a számítására. Ez a másik felület lehet valamilyen sík vagy akár egy másik felületmodell is.

Ha a felületmodellnek vízszintes síkokkal képezzük a metszésvonalait, akkor szintvonalakat kapunk. A szintvonalak generálásakor minden $h = k \cdot a$ magasságú vízszintes síkban elvégezzük a metszetek számítását, ahol a az alapszintköz, k pedig egy egész szám.

Függőleges síkokkal dolgozva a modellezett felület metszeteit tudjuk előállítani. A feladat úgy is megadható, hogy a vízszintes sík egy egyenesének vagy annak egy szakaszának pontjaihoz rendeljük hozzá a terepfelszín magasságát a kérdéses vízszintes helyzetű pozíciókban. Hasonló jellegű feladat a vízszintes sík tetszőleges vonalaival is megfogalmazható, ilyenkor ezeknek a vonalaknak a hossz-szelvényét kapjuk.

Ferde (általános) helyzetű síkokkal vagy egyéb felületekkel is képezhetjük egy felületmodell metszésvonalait. Ilyen vonalakra van például szükségünk, amikor egy földmunkához kapcsolódó töltések és bevágások talp- és körömvonalait szeretnénk meghatározni.

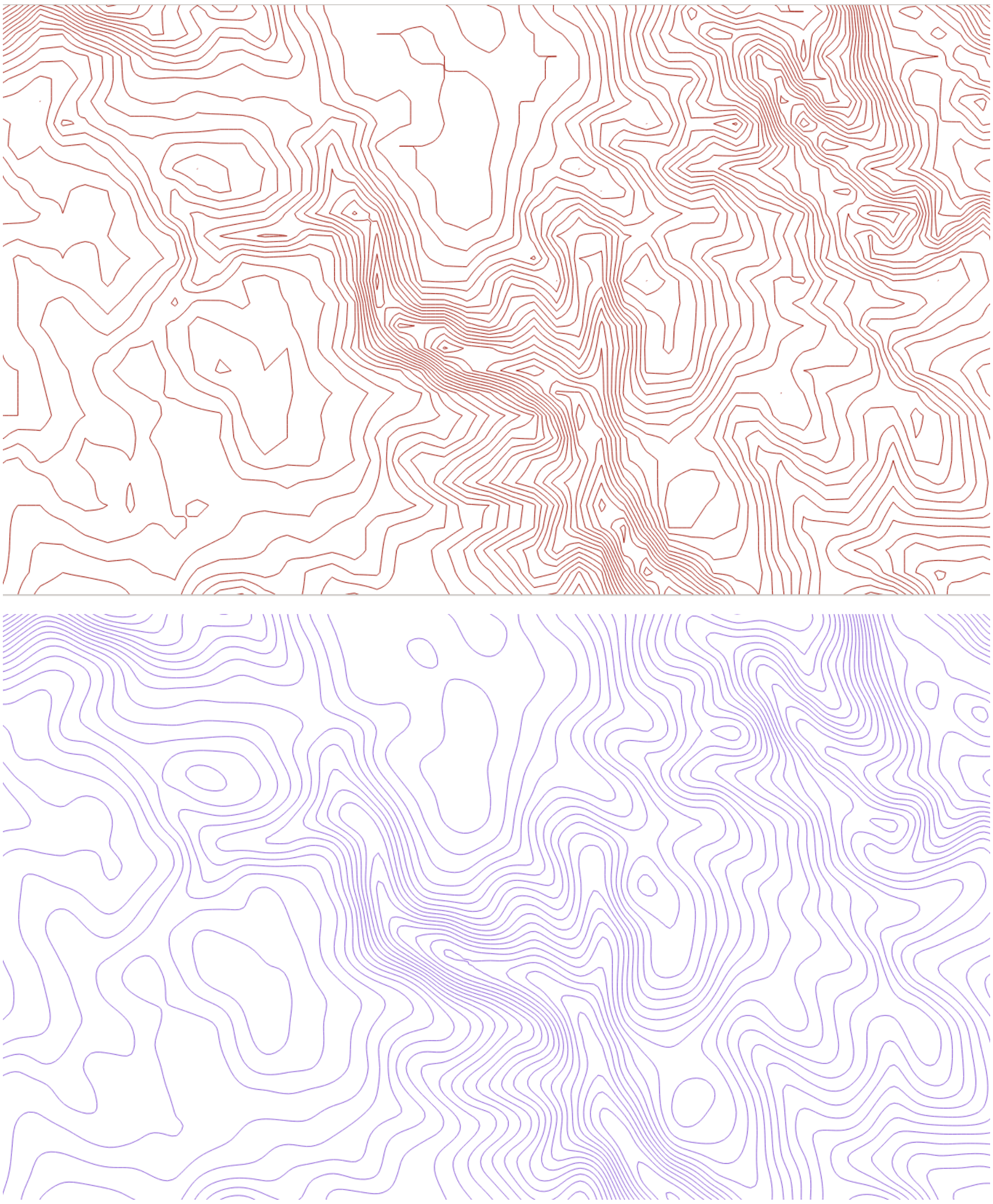
2.6.2. Felületdarab felszínének számítása

Egyes elemzések során szükség lehet a felületmodell egy részletéhez tartozó térbeli felület nagyságának a kiszámítására. A felszín egy elemi darabjának (vagy egy sík tetszőleges méretű részének) felülete $\frac{1}{\cos\alpha}$ szorosa a felszíndarabhoz tartozó síkbeli vetület területének, ahol α a lejtőszög. A felszín egy tetszőleges kiterjedésű darabjának a felülete így a következő módon számítható:

$$A = \iint \frac{1}{\cos\alpha} dxdy = \iint \sqrt{1+p^2+q^2} dxdy = \iint \sqrt{1+\left(\frac{\partial f}{\partial x}\right)^2+\left(\frac{\partial f}{\partial y}\right)^2} dxdy$$

(2.6.1)

A gyakorlatban a kettős integrál számítása helyett a síknak tekinthető alapelemek felületeinek összegzésével dolgozunk. (Ami tulajdonképpen numerikus integrálásnak is tekinthető.) Ilyen módon a felszín egy n darab felületdarabból álló részének összfelülete:



2.5.4. ábra. SRTM alapú (a három másodperces változatból levezetett) domborzatmodellből készített szintvonalak (fenti ábra), majd ugyanennek a felületnek a Catmull-Clark-féle eljárással finomított változatából készített szintvonalak (lenti ábra). A bemutatott eljárás lényege, hogy nem az egydimenziós szintvonalakat próbáljuk meg egymástól függetlenül simítani valamilyen módszerrel, hanem még a szintvonalak generálása előtt a kétdimenziós felületet simítjuk.

$$A = \sum_{i=1}^{n} \frac{T_i}{\cos \alpha_i} = \sum_{i=1}^{n} T_i \sqrt{1 + p_i^2 + q_i^2} \tag{2.6.2}$$

ahol a T_i az egyes felületdarabok vízszintes vetületeinek területei, az α_i pedig a lejtőszögeik. Egységnyi alapterületű (T) elemek esetében (például egy GRID modell esetében) a konstans alapterületnek a szummázás elé történő kiemelésével a következő összefüggés is alkalmazható:

$$A = T \sum_{i=1}^{n} \frac{1}{\cos \alpha_i} = T \sum_{i=1}^{n} \sqrt{1 + p_i^2 + q_i^2} \tag{2.6.3}$$

2.6.3. Térfogatszámítás

A térfogatot a tér egy zárt részére vonatkozóan lehet értelmezni, a felületmodellek pedig csak egy térbeli felületet határoznak meg. A legegyszerűbben úgy tudunk térfogatszámításra alkalmas testeket létrehozni, hogy két felület közötti térrészként határozzuk meg őket, szükség esetén a sík egy felületdarabjával (jellemzően egy poligonnal) lehatárolva.

A gyakorlatban az ilyen jellegű számításokat sokszor földmunkákhoz kapcsolódóan végzik, ahol a két felület a terepnek a munkálatok előtti (eredeti) illetve utáni (tervezett) állapotát írja le. Az ilyen jellegű feladatoknál általában a térfogatszámítás során külön ki kell mutatni a töltések és a bevágások térfogatait, vagyis külön kell összegezni azoknak térrészeknek a térfogatát ahol a változás előtti és ahol a változás utáni állapotot leíró felület van magasabban.

2.7. A domborzat elemzése Fourier-analízis és waveletek segítségével

A domborzatmodellekkel végezhető elemzések és műveletek egy csoportja a Fourier-analízisen alapul. Egy $f(x)$ függvénynek az $\hat{f}(\xi)$ Fourier-transzformáltját az

$$\hat{f}(\xi) = \int_{-\infty}^{+\infty} f(x) e^{-2\pi i x \xi} dx \tag{2.7.1}$$

összefüggéssel tudjuk kiszámítani, a fordított irányba pedig az ehhez nagyon hasonló

$$f(x) = \int_{-\infty}^{+\infty} \hat{f}(\xi) e^{2\pi i \xi x} d\xi \tag{2.7.2}$$

képlettel tudunk dolgozni.

Ezt a transzformációt úgy is fel lehet fogni hogy a tér vagy az idő tartományból a frekvencia tartományba váltunk át, ennek következtében alkalmas lehet elemzési feladatokra valamint akár adatok veszteséges tömörítésére is.

A későbbiekben, a 6.2 részben, is szó lesz egy függvény Fourier-sorral történő közelítéséről. Annak a megoldásnak a közelítés jellege abból adódott, hogy a $-\infty$-től $+\infty$-ig tartó integrál helyett csupán meghatározott számú hullám összegzésével próbáltuk meg helyettesíteni az eredeti függvényt. A 6.2.1 és 6.2.2 képletek látszólag ezen túl is eléggé eltérőek a 2.7.1 és 2.7.2 képletektől de az Euler-képlet ($e^{ix} = \cos(x) + i \sin(x)$) segítségével már felírhatóak azokra jobban hasonlító alakban is.

A Fourier-sorokhoz hasonlóan működik, de azzal nem összekeverendő a diszkrét Fourier-transzformáció (Discrete Fourier transform, DFT). Ilyenkor a korábbiakban látott transzformációt nem függvények, hanem adatsorok között hajtjuk végre. Egy N darab $x_0,\ x_1,\ x_2, \ldots, x_{N-1}$-el jelölt elemből álló adatsornak a szintén N darab, $\hat{x}_0,\ \hat{x}_1,\ \hat{x}_2, \ldots, \hat{x}_{N-1}$-al jelölt elemekből álló Fourier-transzformáltját az

$$\hat{x_k} = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi ikn}{N}} \tag{2.7.3}$$

képlettel lehet kiszámítani. Ezekből az adatokból az eredeti adatsort az

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} \hat{x}_k e^{-\frac{2\pi ikn}{N}} \tag{2.7.4}$$

összefüggéssel kaphatjuk meg. A szükséges számítások az adatsorok elemeinek számának növekedésével négyzetesen növekednek, mert egyszerre növekszik az adatsorok elemeinek a száma, és az egyes elemeket meghatározó összegzések elemeinek száma is.

A 2.7.3 és 2.7.4 képletekben a $e^{-\frac{2\pi ikn}{N}}$ tagok kizárólag az n és a k értékeitől függenek, az adatsorok (x_n-el és $\hat{x}_k$-val jelölt) értékeitől függetlenek. Ezeket az értékeket előre ki lehet számolni és egy $N \times N$ méretű mátrixban el lehet helyezni. A vektornak tekintett adatsorokat ezzel a mátrixszal megszorozva is el lehet végezni a diszkrét Fourier-transzformációt és annak inverzét is. Az ehhez szükséges számítási lépések száma ekkor szintén négyzetesen növekszik az adatsorok méretével.

A gyors Fourier-transzformáció (Fast Fourier transform, FFT) egy olyan számítási megoldást nyújt a diszkrét Fourier transzformációra, amellyel a 2.7.3 és a 2.7.4 képletekben meghatározott számítások $O(N^2)$ lépés helyett $O(N \log N)$ lépéssel is megoldhatóak. Ezt azzal a megkötéssel tudjuk megvalósítani, hogy az adatsorok mérete (N) csak a kettő egész számú hatványa lehet.

A digitális domborzatmodellek vizsgálatához a Fourier-transzformáció kétdimenziós változatára van szükség. Az eredeti adatsort ilyenkor a négyzetrács alapú modell rácspontjainak magasságai képezik, amit általában egy négyzetes mátrixként kezelünk.

A domborzatmodellek Fourier-analízis alkalmazásával történő elemzésével a [131] és a [40] foglalkozik. A [115] az újramintavételezéssel kapcsolatos vizsgálatokban alkalmazza a gyors Fourier-transzformációt.

A Fourier-transzformációnak a veszteséges tömörítésben is fontos szerep jut. A Fourier-transzformációhoz nagyon hasonló diszkrét koszinusz transzformációt (Discrete cosine transform, DCT) alkalmazza (sokféle egyéb módszer mellett) több hang (pl. MP3[12], WMA),

[12] Az MP3 rövidítés az MPEG Audio Layer III megnevezésből ered, a hangformátum az MPEG videók hangjának tárolására kidolgozott kódolás önálló használatából alakult ki.

kép (pl. JPEG[13]) illetve mozgókép (pl. MPEG[14]) tömörítési eljárás; az utóbbi kettő a kétdimenziós változatát. A képtömörítéshez hasonló elven a domborzat veszteséges tömörítésére is alkalmazhatunk a képek tömörítésénél már jól bevált módszereket.

A Fourier-analízisre sok szempontból hasonlít a waveletek alkalmazása. Hullámok helyett itt kisebb, waveleteknek nevezett egységekkel dolgozunk, amikből ahhoz hasonlóan áll össze az eredeti adatsor, mint ahogyan a különféle frekvenciájú hullámokból a Fourier-transzformáció esetében.

Az alkalmazott waveleteket egy ún. anya waveletből (mother wavelet, jelölése $\psi(t)$) kiindulva lehet előállítani. Anya waveletnek olyan függvények alkalmasak, amelyekre teljesülnek a $\int_{-\infty}^{+\infty} |\psi(t)|\, dt < \infty$ és a $\int_{-\infty}^{+\infty} \psi(t)^2\, dt < \infty$ feltételek. Az anya waveletet általában úgy választják meg, hogy az előbbieken túl a $\int_{-\infty}^{+\infty} \psi(t)\, dt = 0$ és a $\int_{-\infty}^{+\infty} \psi(t)^2\, dt = 1$ is teljesüljenek.

A waveletek is alkalmasak képtömörítésre, a JPEG2000[15] szabvány például ilyen megoldást használ. A [44, 52]-ben a waveletek digitális domborzatmodellek tömörítésében, [79]-ben a pedig a geomorfológiai elemzésekben való alkalmazhatóságát vizsgálják.

2.8. Összetett elemzések domborzatmodellekkel

Az alábbiakban néhány példával szeretném bemutatni a domborzatmodellekkel kapcsolatos legfontosabb összetett elemzéseket.

2.8.1. Hidrológiai és egyéb kapcsolódó vizsgálatok

A lehullott csapadék és minden egyéb folyadék a gravitáció következtében lefelé folyik a terep felszínén. Ha egy elemzés során szükségünk van arra, hogy ez a lefele irány merre van és mekkora terepesés tartozik hozzá, akkor ezt egy digitális domborzatmodell segítségével tudjuk megállapitani.

A domborzatmodell ismeretében le tudjuk határolni egy meghatározott ponthoz tartozó vízgyűjtő területet, ahonnan a víz az adott pontba folyik le; vagy el tudjuk különíteni egy terület vízgyűjtő területeit, vagyis vizsgált területnek azokat a részterületeit, ahonnan azonos pontba folyik le a csapadék. (Ezek a pontok lefolyástalan területek mélypontjai, vagy a vizsgált terület határán elhelyezkedő pontok.)

Ezeknek a vizsgálatoknak az alapja az, hogy az egyes elemi felületdarabok lejtésviszonyaiból kimutatjuk, hogy melyik másik felületdarabra folyik át a folyadék az adott területről, így egy irányított gráf segítségével adjuk meg a felületelemek közötti lefolyási viszonyokat. A további elemzéseket az ezen a gráfon végzett műveletekre lehet visszavezetni.

[13] A JPEG rövidítés a formátumot kidolgozó munkacsoport nevéből ered (Joint Photographic Experts Group). A JPEG formátum leírását az ISO/IEC 10918-x szabványsorozat tartalmazza. A munkacsoport honlapja a https://jpeg.org/ címen érhető el. A formátum többféle tömörítési módszert is tartalmaz, de ezek közül a gyakorlatban a DCT-n alapuló veszteséges tömörítést szokták használni.

[14] Az MPEG rövidítés is a formátumot kidolgozó munkacsoport nevéből ered (Moving Picture Experts Group). A többféle szabványt is kidolgozó munkacsoport honlapja a http://mpeg.chiariglione.org/ címen érhető el.

[15] A formátum leírását a ISO/IEC 15444-x szabványsorozat tartalmazza.

A domborzatmodell segítségével ki lehet mutatni, hogy a terepfelszín egyes pontjain milyen mennyiségű víz folyik keresztül egységnyi mennyiségű a területre hulló (egyenletes eloszlású) csapadék hatására. Az ilyen műveletek szoros összefüggésben vannak a talaj erózióját vizsgáló térinformatikai elemzésekkel is.

Fontos megjegyezni, hogy komolyabb vizsgálatokhoz a domborzatmodellen túl még többféle, a terület hidrológiai és egyéb tulajdonságait leíró adat szükséges. Tudnunk kell például, hogy lehullott csapadék mekkora hányada folyik le a felszínen, és mekkora a talajba beszivárgó vagy az elpárolgó víz aránya, illetve az erózió vizsgálatához ismerni kell a talaj jellemzőit is.

2.8.2. Láthatóság vizsgálata

Gyakran felvetődő kérdés, hogy egy pontot látni lehet-e egy másik pontból, másként megfogalmazva a két pontot összekötő térbeli szakasz a terepfelszín alá kerül-e valahol. A gyakorlati alkalmazásokban a szakasz egyik végpontja tekintetében egy rácsháló minden pontjára el lehet végezni ezt a vizsgálatot, az eredmény pedig egy logikai értékeket tartalmazó raszter állomány lesz, ahol az igaz érték a látható, a hamis pedig a takarásban lévő területeket jelöli. A pontok helyzetét a programokban általában vízszintes helyzettel és a terepfelszín feletti magassággal lehet beállítani.

Pontok összeláthatóságának vizsgálatakor fontos kérdés a Föld görbületének figyelembe vétele. Ennek hatása a pontok közötti távolsággal négyzetesen növekszik, így hamar jelentős tényezővé válik.

2.9. Digitális domborzatmodellek térhatású megjelenítése

A digitális domborzatmodellek megjelenítésének egyik gyakran igényelt formája, hogy a domborzatmodell által leírt felületnek a valós arányokkal vagy valamilyen arányú magassági torzítással a térhatású képét állítjuk elő. A térbeli felületen megjelenhet a terület digitális ortofotója, valamilyen térképe, vagy akár a domborzat színfokozatos ábrázolású képe is.

A térhatású megjelenítéshez a domborzatmodell egyes alapelemeit megfelelő transzformációk után a kívánt textúrával (például az ortofotó megfelelő részlete) és megvilágítási jellemzőkkel (a fényforrás vagy a fényforrások helyzete és egyéb jellemzői) kell a számítógépnek kirajzolni. Az egyes alapelemek ilyen módon előállított képeiből áll össze a domborzatmodell térhatású képe.

2.9.1. A digitális domborzatmodell térhatású megjelenítésének eszközei

A térhatású kép előállításának egyik módja az inkrementális képszintézis. Az inkrementális képszintézis során a sugárkövetéses módszerekkel ellentétben a képet nem pixelenként állítjuk elő, hanem nagyobb egységekben, ami jelentős sebességnövekedést eredményez.

A térhatású képet interaktív módon megjelenítő alkalmazások a fenti okból adódóan szinte kivétel nélkül az inkrementális képszintézis elvén dolgoznak, és a térhatású megjelenítés hardveres gyorsítását lehetővé tevő eszközöket is ennek a képelőállítási módszernek a támogatására fejlesztették ki. A képszintézis egységes, a hardver típusától független vezérlése érdekében hozták létre az OpenGL[16] nyelvet, ami tulajdonképpen egy többféle programozási nyelvből is használható felület. [117]

Az OpenGL állapotgép elvén működik. A képszintézis sokféle paraméterét lehet a megfelelő függvényekkel beállítani. Ezek mind befolyásolják a megjelenítendő elemek kirajzolásának eredményét.

Egy domborzatmodell megjelenítése például úgy történhet, hogy a szükséges paraméterek (fényforrások elhelyezkedése és jellemzői, képszintézis egyéb tulajdonságai) beállítása után pontok sorozatát adjuk át egy megfelelő tömbben vagy az erre használható függvények egymás utáni meghívásával. A korábban beállítottak szerint a kapott pontokból hármasával egy-egy háromszögként kirajzolódnak egy TIN modell háromszögei, melyek összességében ki fogják adni a domborzat térhatású képét. Azt, hogy a kapott pontokból hármasával kell egy-egy háromszöget létrehozni szintén egy paraméterként tudjuk előzőleg beállítani; megfelelő igény esetén akár pontnégyesenként egy GRID modell négyszögeit is ki tudjuk hasonló módon rajzoltatni.

Az inkrementális képszintézisről [125] könyvében részletesen olvashatunk. Az OpenGL nyelvnek egy magyar nyelvűre lefordított és kezdők számára is ajánlható bemutatása található [95] könyvében.

2.9.2. A bucka leképezés elve

Ha egy tárgy (ez lehet akár a domborzat is) virtuális másáról jó minőségű képet akarunk készíteni, akkor a modellezés során minél részletesebben kell annak a felületét megadnunk, ami a felszín leírására használt alapelemek számának rohamos növekedésével jár. Ha a felületek egyenetlenségeiből adódó hatásokat is szeretnénk megjeleníteni a képen, az a felület részletességének növelésével csak rendkívül sok háromszög használatával lenne megoldható, ami képszintézis számításigényét is jelentősen megnövelné, teljesen lehetetlenné téve a valós idejű képmegjelenítést.

A probléma megoldását az jelenti, ha a felület apróbb részleteinek modellezését nem a felületdarabok számának növelésével oldjuk meg, hanem egészen más elvet alkalmazunk. Egy felülethez tartozó pixel megjelenítésekor a pixel színét (sok más egyéb jellemző mellett) a fényforrás irányának és a felületre merőleges iránynak (a normálvektornak) az egymáshoz viszonyított helyzete határozzák meg. Amikor egy felület részleteinek megjelenítése érdekében növeljük a modellezett felület alapelemeinek számát, akkor azt azért tesszük, hogy az egyes kis felületdarabokon megfelelő legyen a felület normálvektorának az iránya, így azok a megvilágítás hatására a megfelelő színben (árnyalatban) jelenjenek meg. Ezt viszont úgy is el tudjuk érni, ha egy kevésbé részletes felülethez textúraszerűen hozzárendeljük a felület normálvektorának az eltérését egy részletesebb felület normálvektorától, majd a megjelenítés során a kevésbé részletes felületre számított merőleges irányokat pixelenként korrigáljuk ezzel az értékkel. Az árnyalási egyenlet számításakor

[16] Az OpenGL fejlesztését jelenleg a Khronos Group végzi, a nyelv oldala a https://www.opengl.org/ címen érhető el.

(a pixel színének meghatározásakor) már az így módosított normálvektorokat használjuk fel.

A normálvektor eltérésének megadásához pixelenként két adat is szükséges, így kétféle adatot kell textúra szerűen tárolnunk (normal map). Ha a két felület távolságát tároljuk, abból parciális deriváltként megkaphatjuk az egyszerűsített geometriából számított normálvektor módosításához szükséges két adatot úgy, hogy csak egy adatot kell textúraként tárolnunk. A felületek közötti eltéréseket ilyen módon tároló adathalmazt buckatérképnek (bump map) nevezzük [43].

Földmérőként a bump map és a normal map viszonyát az egyszerűsített illetve a részletes felülethez ahhoz tudnám hasonlítani, ahogyan a geoid unduláció és a függővonal elhajlás viszonyul az ellipszoidhoz illetve a geoidhoz.

A buckatérkép alkalmazása a számítógépes grafika területén elterjedt, mindennaposnak mondható módszernek számít, a képszintézishez kapcsolódó számítások gyorsítását támogató eszközök is kiterjedten támogatják. Ez a támogatás a pixel árnyalónak (pixel shader) nevezett szolgáltatáson keresztül valósítható meg. A pixel árnyaló lehetővé teszi, hogy az árnyaló egység alapértelmezett algoritmusát egy saját kis programra cseréljük le, amivel bucka leképezés támogatását túl még sokféle egyéb dolgot is meg lehet oldani (például a textúrákkal vagy különféle vizuális hatásokkal kapcsolatosan).

Az OpenGL mellett létezik egy külön programozási nyelv, az OpenGL Shading Language [113], amelyben a C nyelvre hasonlító módon tudjuk a pixel árnyalók működését programozni. Ez a nyelv legfontosabb előnyében is hasonlít a C nyelvre: a benne írt programok többféle hardveren is hatékonyan tudnak működni.

2.9.3. Korszerű grafikus eszközök lehetőségeinek kihasználása

Napjainkban már a teljesen átlagosnak mondható számítógépek is nagy teljesítményű, a térhatású modellek megjelenítését támogató grafikus gyorsító szolgáltatásokkal rendelkező eszközöket tartalmaznak. Ezek erőforrásait a digitális domborzatmodellek térhatású megjelenítésekor is ki lehet használni.

A módszer alkalmazható TIN és GRID típusú modellek esetében is. A lehetőség kézenfekvő: készítsük el a domborzatmodell egy egyszerűsített (tehát kevesebb háromszöget tartalmazó, vagy kisebb felbontású) változatát, és állítsuk elő a két felület különbségét, vagyis a buckatérképet. A megjelenítés során így a bucka leképezés elvét tudjuk alkalmazni.

A módszer alkalmazása jelentősen csökkenti az adott minőségű térhatású megjelenítéshez szükséges felületelemek számát, ami által gyorsabbá válik a megjelenítés. Bár a buckatérkép tárolása igényel bizonyos méretű helyet, a felület kirajzolásához szükséges adatok mennyisége még így is jóval kevesebb, mintha a felületelemek számának növelésével próbálnánk hasonló részletességű képet előállítani.

2.9.4. TIN domborzatmodell egyszerűsítésének optimalizálása a bucka leképezést alkalmazó megjelenítés igényei szerint

Az árnyalással jelentkező hatásokat a buckatérkép nagyon jól vissza tudja adni, a kitakarás által okozott hatásokkal viszont nem tud mit kezdeni. Ebből következik, hogy a felület egyszerűsítése során a kitakarás szempontjából fontos részletekre jelentősebb figyelmet érdemes fordítani, az ilyen helyeken kisebb tűrést alkalmazva.

Nagyon fontos kérdés, hogy hogyan származtatjuk az összetett felületből az egyszerűsített felületet. Erre a feladatra többféle algoritmus is létezik, melyek általában a következő két elv valamelyike alapján működnek:

Az első esetben az eredeti felületből kiindulva megkeresik azt a pontot, aminek eltávolítása a legkevésbé változtatja meg a felületet, majd eltávolítják. A műveletet addig ismétlik, amíg kellőképpen le nem csökken a felület elemeinek száma.

A másik esetben kiindulnak egy az eredeti felülettel azonos méretű, azt közelítő elemi felületből, majd azon a ponton, ahol a leginkább eltér a két felület felvesznek egy újabb pontot. Ezt addig ismétlik, amíg az egyszerűsített felület kellőképpen nem hasonlít az eredetire. (De ideális esetben még jóval kevesebb elemből áll)

Bármelyik módszert alkalmazzuk is, a kulcskérdés az lesz, hogy milyen elvek alapján választjuk ki azt a pontot, ahol a két felület leginkább hasonlít, vagy leginkább eltér egymástól. A mi esetünkben a cél a bucka leképezéssel történő megjelenítés szempontjából leginkább megfelelő felület előállítása. Ennek következtében az ezzel a módszerrel létrehozott kép szempontjából leginkább (ha az egyszerű felületből indultunk ki) vagy legkevésbé (ha az eredeti felületből indultunk ki) jelentős pontokat célszerű kiválasztanunk.

A célunk a továbbiakban egy olyan egyszerűsített TIN modell létrehozása, amely kitakarási viszonyai a megjelenítés során jellemző irányokból minél jobban hasonlítanak az eredeti modell kitakarási viszonyaira.

Egy domborzatmodell ábrázolásakor a csúcsok és hegygerincek illetve azok környéke fogják elsősorban eltakarni a mögöttük lévő részeket (mivel a domborzatot jellemzően felülről nézzük), ezért az egyszerűsített felületnek ezeknek a környékén részletesebbnek kell lennie, mint máshol [12]. A terepszerkezeti formák elkülönítésénél, a 6. fejezetben bemutatott módszereket alkalmazhatjuk ilyenkor is a kúpok és a gerincek környezetének megkeresésére. Ezek a területek ott lesznek, ahol a vizsgált hely magasabban helyezkedik el a környezetének átlagos magasságánál. Az egyszerűsítés során ezeken a területeken kisebb tűréssel kell dolgozni.

3. fejezet

Pontfelhőkkel kapcsolatos alapvető ismeretek

A lézerszkenneres mérések a klasszikus geodézia felmérésektől eltérő szemlélettel készülnek, hiszen míg a hagyományos geodéziai felmérésekkor már a terepi munka során elkülönítjük a felmérendő objektumokat és csak néhány meghatározott pontjukat mérjük fel egyenként, kis méretű, de már jól strukturált adathalmazokat létrehozva; addig a lézerszkenneres mérések során a műszer rengeteg pontot mér fel egy meghatározott térbeli kiosztásban, nagy méretű, de komoly további feldolgozást igénylő adathalmazt, a pontfelhőt eredményezve. A pontfelhő nem csak a felmérés technológiája szempontjából, hanem a tárolás és a feldolgozás módszereit tekintve is teljesen más megoldásokat igényel, mint a klasszikus geodéziai mérések.

Ebben a fejezetben a pontfelhőkkel kapcsolatos alapvető ismereteket szeretném bemutatni. Ennek keretében a pontfelhők létrehozására alkalmas technológiákkal és tárolásuk, feldolgozásuk valamint megjelenítésük kérdéseivel is foglalkozom. Mindezek ismertetése közben lézerszkenneres felmérésekkel és a pontfelhők feldolgozásával kapcsolatos, önálló tézisnek nem minősíthető kutatásaimat és egyéb munkáimat is bemutatom.

3.1. Pontfelhők létrehozása

Pontfelhők általában lézerszkenneres felmérések során keletkeznek. A lézerszkenneres mérések alkalmával nagyon nagy számú lézeres távolságmérést végzünk különféle irányokba; a lézersugár irányítását ilyenkor valamilyen forgó tükör vagy egyéb forgó alkatrész végzi úgy, hogy a lézersugár irányát pontosan ismerjük a műszer koordinátarendszerében. A lézeres távméréssel meghatározott távolságot is felhasználva így a felmért pontok térbeli helyzetét számítani lehet a műszer koordináta-rendszerében.

A lézeres távmérés alapelve, hogy a fény kétszer járja meg a megmérendő távolságot amíg eljut egy felületre, majd az ott szétszóródó fény egy kicsiny hányada visszajut a műszerbe. A távolság meghatározása történhet közvetlenül az idő mérésével, vagy megvalósítható a lézer fényforrást modulálva fázismérés elvén is. Mindkét módszer esetében rögzíteni szokták a távolság mellett a műszerbe visszatérő fény erősségét is.

A műszer környezetének letapogatásához a lézersugarakat két dimenzióban is irányítani kell; a felmérés közben mozgó lézerszkennereknél (a földi járműre szerelt mobil, illetve a repülő járművön elhelyezett légi lézerszkennerek) az egyik dimenziót általában a hordozó jármű mozgása biztosítja. Földi lézerszkennereknél mindkét irányról a műszernek kell gondoskodnia, ami általában a teodolitokhoz hasonló módon egy állótengely körüli forgással és egy tükörrendszernek egy a fekvőtengelynek megfeleltethető tengely körüli forgatásával valósul meg.

Mint minden mérésnél, itt is fontos, hogy ismerjük a műszereink pontosságát és megbízhatóságát. Ezzel kapcsolatban a hazai szakirodalomban a [37, 36]-ban olvashatunk részletesen.

A mérések feldolgozása szempontjából fontos, hogy a felmért pontokat műszer koordináta-rendszeréből átszámítsuk egy megfelelő vonatkozási rendszerbe. Ez a rendszer akár egy helyi koordináta-rendszer is lehet, például az egyik álláspont koordináta-rendszere vagy egy a felmért létesítmény által kijelölt rendszer; de általában a pontfelhőt egy geodéziai koordináta-rendszerben kell elhelyezni, georeferálni. A földi lézerszkenneres méréseknél az egy álláspontban való mérés során a műszer mozdulatlan marad, így a műszer koordináta-rendszere és a mérések feldolgozásához használt koordináta-rendszer kapcsolatát leíró transzformáció paraméterei egy állásponton belül állandóak, ezzel a transzformációval az adott álláspontban mért pontfelhő egyszerűen átszámítható a feldolgozás során használt rendszerbe.

A különféle álláspontok közötti geometriai kapcsolatot többféle módon is meg lehet teremteni. Ezeknek a módszereknek az egyik csoportja közös pontok mérésén alapul, ami azért körülményes kissé, mert a lézerszkennerrel nem lehet a mérőállomásoknál megszokott módon pontokat megirányozni és meghatározni; ehelyett olyan jeleket (target) kell elhelyezni, amelyeknek pontosan meghatározhatjuk a képződött pontfelhő alapján a középpontjukat. Vannak felmérési technológiák, ahol egyenként meg kell jelölni (a műszerbe épített kamera vagy munkaterület előzőleg beszkennelt pontfelhője segítségével) és azonosítóval kell ellátni a használni kívánt jeleket, amiket utána a műszer nagy felbontással külön is beszkennel. Ez nagy pontosságot és megbízhatóságot jelent, de a felmérés idejét jelentősen meghosszabbítja.

Más módszereknél a munkaterület felmérésével együtt beszkennelt jelekkel dolgozunk, így a felmérési munka gyorsabban és egyszerűbben végezhető. A feldolgozóprogram a felmért pontfelhőkben automatikusan megkeresi a jeleket és meghatározza középpontjaikat. A jeleknek ilyenkor azonosítót sem kell adni, csak megfelelő körültekintéssel ki kell helyezni őket a mérés előtt, mert a program az elhelyezkedésük alapján azonosítani tudja őket és az egyes álláspontokban végzett méréseket így puzzle-szerűen össze tudja illeszteni.

Jelnek sokféle eszköz alkalmas lehet. Ezek egyik csoportjának alapelve az, hogy egy sík felületre festenek fel valamilyen jól meghatározható középpontú ábrát például egy korongot vagy két egymáshoz a sarkával érintkező négyzetet. A síkra felvitt ábrát a visszajövő jel erőssége (az intenzitás) alapján lehet elkülöníteni a környezetétől. Az ilyen jelek készülhetnek matricaként vagy egyéb felragasztva rögzíthető formában, vagy elhelyezhetik az őket tartalmazó lemezt egy olyan eszközön, amelyikkel azt a jel középpontja körül lehet tetszőleges térbeli irányba forgatni, amire azért van szükség, mert az ilyen típusú jeleket csak egy bizonyos szögtartományban lehet használni a síkjukra állított merőlegeshez képest. A jelek egy másik csoportja valamilyen jól meghatározható középpontú térbeli geometriai elem, például egy gömb. A gömb jelek előnye, hogy minden irányból egyformán jól mérhetőek anélkül, hogy irányba kellene őket forgatni.

Vannak módszerek, amelyekkel jelek kihelyezésére nincs szükségünk, mert az egyes álláspontokat a felmért pontfelhők átfedő részei alapján is össze lehet illeszteni. Ezt a feldolgozóprogramok különféle mértékben támogatják, van ahol a kezdő illesztést a felhasználónak kell elvégeznie néhány közös pont megadásával, de akár teljesen automatizáltan is felismerheti a program a pontfelhők átfedő részeit.

Egyes műszerek a mérőállomásokban használtakhoz hasonló kompenzátorral rendelkeznek. Ennek segítségével biztosítható, hogy a műszer koordináta-rendszerének XY síkja a vízszintes sík legyen. Az álláspontok koordináta-rendszereinek illesztéséhez ilyenkor két közös pont is elég az egyébként szokásos három helyett.

Ha a földi lézerszkennelés eredményét egy geodéziai koordináta-rendszerben szeretnénk elhelyezni, akkor olyan pontokra is szükségünk van, amelyeknek ismerjük a koordinátáit ebben a rendszerben. Ezek lehetnek olyan, a pontok illesztésénél is használt jelek, amelyeknek meghatározzuk valahogyan a koordinátáit, de bármilyen más, a pontfelhőben a kívánt pontossággal beazonosítható pont megfelel. Magának a műszernek a helyzetét is meg lehet határozni (például pontra állva a műszerrel, vagy a műszert kényszerközpontosan kicserélve más eszközre, esetleg a műszeren mérés közben elhelyezhető eszköz segítségével), így egy álláspont koordináta-rendszerének kezdőpontját tudjuk majd megadni a kérdéses geodéziai koordináta-rendszerben.

Mobil és légi lézerszkennereknél a földi lézerszkennerekkel ellentétben a műszer koordináta-rendszere a hordozó járművel együtt folyamatosan mozog a mérések során, így a mérések georeferálásához szükséges transzformáció paraméterei is folyamatosan változnak az idő függvényében. A folyamatos változás következtében így folyamatosan meg kell tudnunk határozni megfelelő pontossággal a műszer helyzetét és irányát. A mobil és légi lézerszkennereknek ezért elválaszthatatlan részét képezik a helymeghatározó (általában GNSS alapú) és inerciális rendszerek. Az ezekkel végzett méréseknek köszönhetően a felmért pontfelhő georeferálható, további (például megfelelő jelekre végzett) mérésekre csak legfeljebb a pontosítás vagy az ellenőrzés miatt lehet szükség.

Egyes fotogrammetriai kiértékelések eredménye is lehet pontfelhő [141, 86]. Ilyenkor a fényképfelvételeket felvételpáronként összehasonlítva keresik meg az illeszkedő részeket [29, 110], majd azok képeken belüli elhelyezkedéséből megállapítják a lefényképezett objektumrészlet térbeli helyzetét. Ha egy objektumról több képet készítünk, azokból sokféle alkalmas felvételpár állítható össze (n darab felvétel esetében elméletileg $\frac{n(n-1)}{2}$ féle párosítás lehetséges, ezek közül értelemszerűen csak az lesz használható, ahol a felvételpár képeinek jelentős része ugyanazt a területet ábrázolja), majd az egyes felvételpárokon is sok helyen lehet illeszkedő részeket keresni. Minden ilyen illeszkedés a tér egy-egy pontját határozza meg, amelyek összességükben egy pontfelhőt alkotnak.

Az illeszkedő részek keresése kétféle elven történhet. Az egyik módszer szerint a felvételpár két képén keresünk egymáshoz illeszkedő részeket; egy másik, főként légifelvételek alapján történő felszínmodell kiértékelésre használt eljárás szerint különféle feltételezett magasságok esetére vizsgáljuk meg, hogy az adott magasságokhoz rendelhető képrészletek (amelyek akkor tartoznának a képeken az adott vízszintes helyzetű ponthoz, ha a terep magassága ott a feltételezett érték lenne) mennyire illeszkednek. A vizsgálatok során általában a 4.1 részben bemutatott piramis index elvéhez hasonló módon kisebb felbontású változatokat készítenek a felvételekről, majd a kisebb felbontástól a nagyobb felé haladva végzik el az egyre részletesebb és pontosabb illesztéseket [46, 65].

Korszerűbb programok lehetővé teszik a fentiekben bemutatott folyamat teljes automatizálását. Ennek köszönhetően a felhasználónak csak meg kell adnia az adott területről vagy objektumról készült felvételeket, majd a program minden további lépést elvégez: megkeresi a kiértékeléshez alkalmas képpárokat, majd elvégzi azokon a kiértékelést felhasználói közreműködés nélkül úgy, hogy közben még a felvételek térbeli elhelyezkedését (a kamera helyzete és iránya, valamint amatőr kamera esetén a belső tájékozás egyes paraméterei is) is meghatározza [54, 100].

A pontfelhő sűrűsége általában nem egyenletes. Földi felmérés során az egyes álláspontokhoz közeledve, mobil szkenner esetében a jármű útvonalának közelében sűrűbben helyezkednek el a pontfelhő pontjai, mint azoktól távolabb. A légi lézerszkenneres felmérések pontfelhői egyenletesnek mondhatóak, legfeljebb a különböző sávok közötti átfedésekhez kapcsolódóan tapasztalhatók sűrűsödések, illetve a függőleges felületeken (épületek homlokzatai) ritkulások.

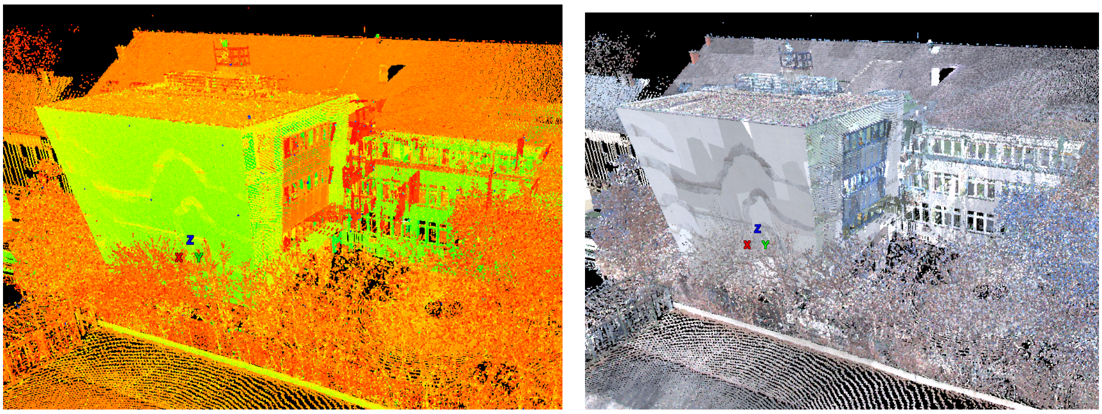
3.2. Pontfelhők pontjainak jellemzői

A pontfelhő több millió vagy akár milliárd pontból áll, amelyek összességükben adják a felmért terület nagy részletességű reprezentációját. A pontfelhő lehet rendezett, amikor a felmérés során felvett sorok és oszlopok alapján egy kétdimenziós tömbben tároljuk pontokat, vagy lehet rendezetlen is.

Az egyes pontokat a helyzetükön túl még jellemezheti egy a távmérés során visszatérő jel erősségétől függő szám, amit intenzitásnak (intensity) szokás nevezni. Az, hogy ezek az intenzitás értékek milyen tartományba esnek és pontosan milyen értékeket vehetnek fel műszerenként illetve alkalmazásonként változhat.

A pontfelhő pontjainak fontos kiegészítő tulajdonsága lehet egy RGB számhármassal megadható szín, amit valamilyen a felmérés során készített fényképfelvétel alapján lehet meghatározni. A pontfelhő kiszínezésére használt fényképfelvételek készülhetnek a távméréssel azonos pontból (pl. az egyes földi lézerszkennereken, például a Leica ScanStation C10-ben alkalmazott kamerával kombinált forgótükrös megoldások segítségével) vagy külpontosan is. A pontfelhő pontjainak kiszínezése pontosabb és egyszerűbb is az azonos pontból készült felvételeknél, mintha a kamera külpontosan helyezkedne el. Fotogrammetriai módszerrel, felvételpárok illesztésével nyert pontfelhők esetében a szín meghatározása kézenfekvő módon történik a felhasznált felvételek alapján.

További fontos kérdés, hogy a fényképfelvételek készítése időben mennyire tér el a lézerszkenneres mérésektől. Egyes mobil felmérő rendszereknél ez akár másodperc alatti is lehet, mivel a járművön elhelyezett kamerák adott időközönként folyamatosan fényképeznek, miközben a lézerszkenner(ek) is folyamatosan dogozik. Földi lézerszkennereknél a szkennelés és a fényképezés általában külön munkafázisokban történik, ami a mérések idejének megfelelő mértékű időeltérést eredményez. Bizonyos esetekben előfordulhat, hogy egy időben teljesen eltérő, akár évekkel korábban vagy később készült felvétel színeit illesztjük a pontfelhő pontjaira, például ha egy LiDAR felmérés pontjait egy digitális ortofotó alapján színezzük ki. Az időbeli eltérés nem kívánt következménye,

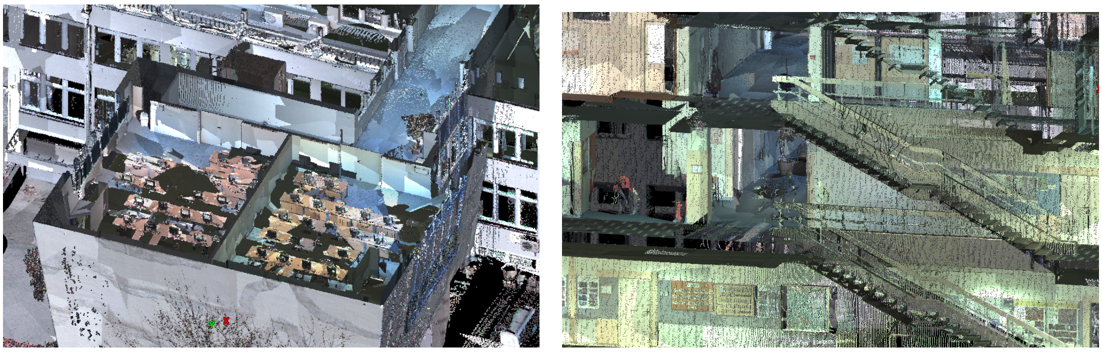


3.2.1. ábra. Pontfelhő megjelenítése intenzitások alapján (bal oldalon) illetve a fényképfelvételekről átvett színekkel színezve (jobb oldalon). A képeken szereplő pontfelhő a szerző és tanítványai által az ÓE AMK Geoinformatikai Intézetének Pirosalma utcai épületében illetve annak környékén Leica ScanStation C10 műszerrel végzett méréseknek Leica Cyclone szoftverrel történő feldolgozásával készült.

hogy a fényképen és a pontfelhőn eltérő dolgok képződnek le, így a pontfelhő nem megfelelően lesz kiszínezve. Például, ha valaki a fényképek készítésének fázisában a kamera elé kerül, annak a képe felkenődhet a közeli falra, fordított esetben pedig a lézerszkenner által beszkennelt személyhez tartozó pontfelhő pontjai a fallal megegyező színűek lesznek.

A pontfelhő egyes pontjaihoz egy merőleges irányt is hozzá szokás rendelni. Ez geometriai szempontból kissé furcsának tűnhet, hiszen egy pontra nem lehet merőlegest állítani, de ha abból indulunk ki, hogy a pontfelhő egy pontja egy felülethez tartozik, már könnyen megláthatjuk a dolog lényegét. A pontfelhő egy pontjának a „merőlegese" úgy határozható meg, hogy a pontra és a szomszédos pontokra egy síkot vagy más alkalmas felületet illesztünk, majd ennek a merőleges (normális) irányát vesszük. A szomszédos pontok rendezett pontfelhő esetében egyszerűen adódnak, rendezetlen pontfelhőknél a térben legközelebbi pontokat lehet kiválasztani. A merőleges irányra többféle elemzésnél is szükségünk lehet.

Azonosító jellegű tulajdonságokat a pontfelhő pontjaihoz nem szokás külön hozzárendelni, mivel azok önmagukban nem rendelkeznek semmiféle azonosításra szükséges jellemzővel. Mint azt már az 1.4 alfejezetben kifejtettem, ellentétben a klasszikus geodéziai felmérésekkel, amikor a meghatározott pontok közvetlenül valamilyen objektum alakjelző pontjai (például egy épület sarka) és emiatt az egyértelmű azonosításuk nyilvánvalóan szükséges, a pontfelhő pontjai összességükben nyújtanak egy modellt (hasonlóan egy fénykép pixeljeihez), így az egyes pontok egyenkénti beazonosítása nem annyira fontos. Nyilván a pontfelhőt kezelő program azonosítani tudja valahogyan az egyes pontokat, ha mást nem, akkor a pontfelhő tárolása során adódó egy- vagy (rendezett pontfelhő esetén) kétdimenziós indexszel.



3.3.1. ábra. Pontfelhő megjelenítése vízszintes (bal oldal) illetve függőleges (jobb oldal) metszet segítségével. A felhasznált pontfelhő megegyezik a 3.2.1 ábrán láthatóval.

3.3. Pontfelhők megjelenítése

A pontfelhők kezelésére és megjelenítésére külön eszközre van szükség az egyes GIS, CAD vagy 3D modellező programokon belül. Bár megfelelő számú pont objektum segítségével is leképezhető egy pontfelhő, de ennek hatékonysága ahhoz hasonlítható, mintha egy raszter képet különféle színű négyzet alakú poligonokkal próbálnánk meg pixelenként kezelni.

Ha a pontfelhő pontjait a képernyő geometriai értelemben hozzájuk tartozó pixeljének előre meghatározott színűre történő színezésével próbálnánk ábrázolni, akkor az egy részleteiben nehezen elkülöníthető pacaként jelenne meg. Ehhez hasonlót a 8.3.3 ábrán láthatunk, ahol az illeszkedő gömbök animálása miatt használtam egy olyan programot, ami a továbbiakban bemutatott ábrázolási módszerekre nem volt képes.

A pontfelhő megjelenítésekor az egyes pontokat vagy a ponthoz külön eltárolt színnel (ha rendelkezésre áll ilyen) vagy pedig egy színskála alapján az intenzitásból származtatott színnel szokás megjeleníteni. Ez utóbbi megoldás előnye, hogy a hozzá szükséges adatok a távméréssel együtt létrejönnek, így használata nem igényel semmiféle járulékos munkát.

A pontfelhő egyes pontjait a hozzájuk rendelt merőleges irány segítségével lehetőségünk van árnyaltan megjeleníteni. Ilyenkor nem a tárolt vagy az intenzitásból származtatott színnel közvetlenül lesz az adott pixel kiszínezve, hanem egy árnyalási egyenlet ([125] 123. oldal) segítségével a ponthoz tartozó normális irányát is figyelembe vesszük. A megjelenített kép így sokkal plasztikusabb lesz, a pontfelhő térbeli formája sokkal jobban kivehetővé válik.

A pontokhoz rendelt normális irányokat akkor is fel tudjuk használni, ha egy belülről beszkennelt terem megjelenítésekor el szeretnénk tüntetni a belátást akadályozó falakat. Ilyenkor nem jelenítjük meg azokat a pontokat, amelyek esetében a pont normálvektorának és a kamera tengelyvonalához rendelt irányvektornak a skaláris szorzata pozitív szám.

Az egyes szoftverek a megjelenítés során még sokféle lehetőséget nyújthatnak, lehetővé tehetik például különféle metszetek, szeletek vagy térbeli kivágatok készítését. Sok programban lehetőségünk van animációk készítésére is úgy, hogy a kamerával egy adott útvonal mentén haladva tudjuk bejárni a pontfelhőt.

3.4. Geometriai műveletek a pontfelhővel, mint felülettel

Az alábbiakban olyan módszereket mutatok be, amelyek segítségével a pontfelhőt mint felületet tudjuk kezelni különféle geometriai műveletek során. Ehhez a felülettel végezhető geometriai művelteket kissé általánosítani kell, hogy pontok halmazával is elvégezhetőek legyenek.

3.4.1. Pontfelhő és sík metszésvonalának meghatározása

A pontfelhő pontok halmazából áll, amelyek viszont alapvetően egy felületet írnak le. A pontfelhőből sokféle módon eljuthatunk ehhez a felülethez. Vannak esetek, amikor egy későbbi lépésben az így kapott felületnek a metszésvonalát szeretnénk előállítani egy síkkal vagy más megadott felülettel. Az alábbiakban bemutatok egy módszert, amivel ezt a feladatot közbenső illesztett felület előállítása nélkül oldhatjuk meg.

Elsőként a pontfelhő pontjaiból egy β-vázat (beta skeleton) kell előállítani. Ez egy olyan gráf, amelynek csúcsai a pontfelhő pontjai, amelyeket akkor köt össze egy él, ha nincs a pontfelhőnek olyan harmadik pontja, amelyikből a két pont egy θ-val jelölt szögnél nagyobb szög alatt látszik. A θ a váz nevét adó β paraméter függvénye, értéke $\beta \geq 1$ esetében $\arcsin \frac{1}{\beta}$, $\beta \leq 1$ esetében pedig $\pi - \arcsin \beta$. A $\beta = 1$ esetben bármelyik képlettel számolva a $\theta = \frac{\pi}{2}$ értéket kapunk. A β-vázról, annak különféle alkalmazási lehetőségeiről, illetve a előállítását lehetővé tévő algoritmusokról a [57] tartalmaz bővebb ismereteket[1].

A β-váznak a $\beta = 1$ esetét, amikor a θ derékszög, Gabriel-gráfnak (Gabriel graph) [64] is szokták nevezni. Ebben az esetben akkor kötjük össze egy éllel a pontfelhő két elemét, ha a közéjük, mint egy átmérő két végpontja közé illesztett gömbben nincsen a pontfelhőnek más pontja, hiszen a gömb belsejében elhelyezkedő pontban kaphatnánk csak a θ-ra derékszögnél nagyobb szöget. Egy másik megközelítésben ez azt jelenti, hogy a pontfelhő két pontját akkor kötjük össze egy éllel, ha nincs a pontfelhőnek olyan harmadik pontja, amely esetében a két vizsgált ponttól mért távolságok négyzeteinek összege kisebb, mint a vizsgált pontok közötti távolság négyzete. A távolságok négyzeteit a koordinátakülönbségek négyzeteinek összegzésével egyszerűen lehet számítani. A síkban az így kapott gráf a Delaunay háromszögháló gráfjának részgráfja. A témával (más témák, például egy γ-gráfnak nevezett struktúra bemutatása mellett) részletesen foglalkozik [136, 135].

A β-váz előállításakor nem egy felületet, hanem csupán egy egyenes szakaszokból álló vázat kapunk. Ha ezt a vázat elmetsszük egy síkkal (vagy más felülettel), akkor az eredmény nem egy vonal, hanem metszéspontok halmaza lesz. A kapott ponthalmazból viszont lehetőségünk van ismét egy β-vázat készíteni, amelynek vonalai már megfeleltethetőek lesznek egy metszésvonalnak.

Több, egymással párhuzamosan felvett síkokkal képzett metszésvonalakból akár felületeket is elő lehet állítani.

[1] A β-vázról a https://en.wikipedia.org/wiki/Beta_skeleton címen is lehet ismereteket szerezni.

3.4.2. Pontfelhő és egyenes döféspontjának meghatározása

Döféspontnak egy egyenes és egy felület közös pontját nevezzük. A pontfelhő által meghatározott felületnek a metszésvonalhoz hasonlóan a döféspontját is meg lehet határozni közbenső felület meghatározása nélkül.

A klasszikus geometriai feladatnak megfelelően ezt az esetet is vissza tudjuk vezetni vezetni egy a döfő egyenesre illesztett tetszőleges sík és a vizsgált felület metszésvonalának előállítására, majd ennek a metszésvonalnak a döfő egyenessel való visszametszésére. Mivel a metszésvonal nem egyenes, a döfő egyenest akár többször is elmetszheti, így több döféspont is keletkezhet.

A négyzetrács-szerű kiosztásban egymással párhuzamosan elhelyezett egyenesekkel képezett döféspontok szintén alkalmasak lehetnek egy felületmodell előállítására. A pontfelhőt megjelenítő képernyő egy pontjához is hozzárendelhetünk egy (a nézeti transzformáció alapján kiszámítható térbeli helyzetű) félegyenest, majd meghatározhatjuk az ahhoz tartozó (a félegyenes kezdőpontjához legközelebb elhelyezkedő) döféspontot. Ezt az elvet alkalmazva a felhasználó rajzolni tud a pontfelhő felületére.

A pontfelhők feldolgozására már [83] is javasolta az előzőekhez kapcsolódóan bemutatott gráfokat, de ebben a cikkben a közvetlenül a felületek illesztésére koncentrálnak, míg az általam javasolt módszerek közvetlenül a pontfelhőből, közbenső felület létrehozása nélkül dolgoznak.

3.5. Gyakorlati példák pontfelhők alkalmazására

A lézerszkenneres felmérésnek és ehhez kapcsolódóan a pontfelhők feldolgozásának a legkülönfélébb területek és objektumok felmérésében jut szerep. A technológia mindenhol eredményesen alkalmazható, ahol egy területről részletes térbeli modellt akarunk készíteni. A felmérés és a feldolgozás eszközei a konkrét feladattól illetve annak paramétereitől függenek.

3.5.1. Terepmodell készítése pontfelhők alapján

A digitális terepmodell a digitális domborzatmodellből és a terepet alkotó objektumok (nagyjából a topográfiai térkép síkrajzi része) megfelelő térbeli modelljeiből áll. A digitális terepmodell pontfelhő alapján történő előállításakor tehát elő kell állítani a domborzatmodellt, valamint fel kell ismerni és ki kell értékelni a pontfelhőben leképződött egyéb objektumokat.

A digitális terepmodell lehet a hagyományos topográfiai térkép egy digitális változata, de annál akár részletesebb adathalmaz is készíthető. Ilyen lehet például egy olyan városmodell, amely már az egyes épületek különféle részletességű térbeli (3D) modelljeit is tartalmazza. Az épületek és egyéb a modellt alkotó objektumok esetében a CityGML[2] szabvány például LOD0 és LOD4 között öt különféle részletességi szintet határoz meg. A LOD a Level of Detail kifejezés rövidítéséből ered.

A digitális terepmodell kifejezést egyes helyeken a digitális domborzatmodell szinonimájaként is használják.

[2] A CityGML szabvány a http://www.opengeospatial.org/standards/citygml oldalon érhető el.

3.5.2. Épületek és épített környezet felmérése

A lézerszkenneres felmérési technológia kiválóan alkalmas épületek külső és belső tereinek felmérésére. A belső terek felmérésekor a földi lézerszkennerek jöhetnek számításba, azok közül is főként azok a műszerek, amelyek a teljes körülöttük lévő teret képesek beszkennelni, horizontálisan minden irányban, és vertikálisan is csak a műszer alatt elhelyezkedő térrészt (A Leica ScanStation C10 esetében például a -45 fokos magassági szög alatti részek) kihagyva. Az épületek külső felmérésekor az előbb bemutatottakon túl már számításba jöhetnek nagyobb területek esetén a mobil lézerszkennerek illetve a homlokzatok felmérésére az olyan földi műszerek is, amelyek csak korlátozottabb tartományban képesek dolgozni.

Egy épületnek a külső és a belső felületeit egyaránt és akár egyszerre is fel lehet mérni, így azok együttesen az épület nagy részletességű térbeli modelljét adják. (lásd a 3.2.1 és a 3.3.1 ábrák képein) Az ilyen modellek nagyon jól használhatóak építészeti vagy műemlékvédelmi [91] célokra, hiszen a rendkívül részletesen és pontosan dokumentálják a felmért épületet, és ezáltal egy az épületet érintő átalakítás tervezéséhez is kiváló alapot nyújtanak.

Az épületek felmérése irányulhat egy épület helyett egy nagyobb területre, egy háztömbre vagy akár egy kisebb városrészre is. A [24, 13] egy ilyen munkát mutatnak be. Nagyon fontos szerepe lett a lézerszkenneres technológiáknak a mérnökgeodéziában [90] is, és a régészeti célú alkalmazásuk [30] is egyre jelentősebb.

3.5.3. Barlangok felmérése

A barlangokat alkotó üregek alakja teljesen szabálytalan, a barlangot kialakító természeti erők egészen változatos formákat hozhatnak létre. A barlang belső felületének pontos és részletes felmérésére kézenfekvő lehetőségként adódik a lézerszkenneres technológia [127].

Földi lézerszkennerek segítségével több hazai barlang (elsősorban a kiépített barlangok) egyes részeit felmérték már. A szerző ezek közül a mérések közül a Pál-völgyi-barlangban [14] (3.5.1. ábra), a Szemlő-hegyi-barlangban valamint a Béke-barlangban [15] végzett mérésekben vett részt.

Mobil lézerszkennerek a barlangok felmérése során több okból sem alkalmazhatóak. Az egyik ilyen ok, hogy zárt terekben a helymeghatározás körülményes, bár ez még áthidalható fejlett inerciális helymeghatározó rendszerek segítségével, amiket alagutak szkennelésekor alkalmazni is szoktak. A másik ok, hogy a mobil lézerszkennert hordozó jármű közlekedése nem biztosítható egy barlangban. (Sokszor még a földi szkenner szállítása és használata is körülményes a barlangokban végzett mérések során.)

Barlangok felmérése során a föld alatti fényviszonyok nem teszik lehetővé a fényképek készítését a lézerszkenneres felmérési technológiáknál szokásos kamerákkal. A barlangokban készített lézerszkenneres mérések pontfelhőit ezért elsősorban az intenzitás alapján lehet megjeleníteni. (lásd 3.5.1 ábra)

A barlangok felmérése során keletkező adatokat általában a szélesebb közönség számára is elérhetővé szeretnénk tenni. Ilyenkor jól jönnek az olyan külön program telepítése nélkül webböngészőből használható megoldások, amelyekre egy példát a [15] mutat

3.5.1. ábra. A szerző 2012 májusában, a Pál-völgyi-barlangban végzett lézerszkenneres mérések [14] során egy Leica ScanStation C10 földi lézerszkennerrel (bal oldal); valamint a fényképen láthatóval nagyjából megegyező terület képe a mérésekből kapott pontfelhőben az intenzitások alapján színezve (jobb oldal).

be.

3.5.4. Egyéb alkalmazási lehetőségek

A földi lézerszkennert számos esetben alkalmazhatjuk, amikor valamilyen tárgynak a pontos alakját kell gyorsan és részletesen felmérni. Akár tárgyszkenner helyett is lehet használni, ha a modellezni kívánt dolog nem fér be a tárgyszkennerbe, vagy ha nincs ilyen eszközünk. Részt vettem már olyan munkákban, ahol kartográfiai örökség archiválása céljából mértünk fel dombortérképeket [16].

Különféle természettudományi kísérletek eszköze is lehet a földi lézerszkenner. A Debreceni Egyetemmel együttműködve 2013-ban földi lézerszkenneres méréseket végeztem a Természetföldrajzi és Geoinformatikai Tanszék folyóvizes laboratóriumában, ahol egy a víz eróziós hatását modellező terepasztal ismételt felmérése volt a feladatom a kísérletek során [10], aminek segítségével így a különféle fotogrammetriai módszerek [39] mellett lézerszkenneres technológiával is követni lehetett a terepasztal felszínének változásait.

A pontfelhőknek és a lézerszkenneres méréseknek a jövőben előreláthatólag fontos szerep jut majd a precíziós mezőgazdaságban is. [111]

4. fejezet

Domborzatmodellek tárolása során használható indexelési módszerek

A digitális domborzatmodell esetében, akár csak más adatoknál, a hatékony tárolásra törekszünk. A hatékonyság többféle dolgot jelenthet.

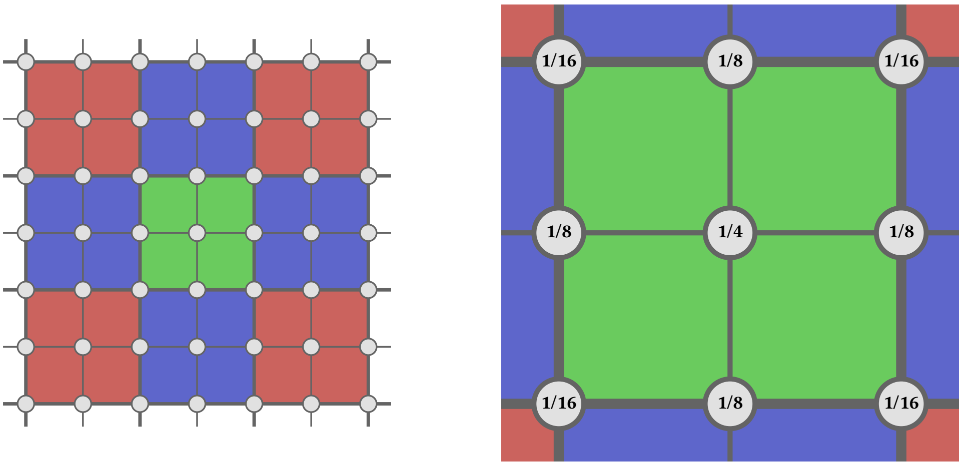
Hatékonynak tekinthetjük azt az adatszerkezetet, amiből kiindulva egy megfelelő algoritmus kevés (vagy az adathalmaz méretével kevésbe növekvő) számítási lépéssel elő tudja állítani a kívánt eredményt. A hatékonyság egy másik mutatója a domborzatot leíró adathalmaz mérete, itt nyilvánvalóan a kisebb méretet tekintjük hatékonyabbnak. Végezetül meg kell említeni, hogy sok esetben az adathalmaz egyszerűsége is nagyon fontos előny.

A fenti szempontokat egyszerre kielégíteni nagyon nehéz, mivel azok sokszor egymással ellentétesen hatnak. Például egy tömörítést lehetővé tévő adatformátum alkalmazása az adathalmaz méretének csökkenése mellett azzal jár, hogy az egyes műveletek előtt dekódolni kell a tömörített adathalmazt vagy annak legalább egy (a vizsgált területre eső) részét. Számos esetben hatékonyan csökkenthető egy a domborzatmodellen végzett művelet számítási igénye a modell alapelemeinek térbeli indexelésével, de a tárolandó adathalmaz mérete az index méretével növekedni fog. Ezeken a szempontokon túl még azt is figyelembe kell venni, hogy az adathalmazunk mindkét előző példában bonyolultabbá is válik a tömörített adatok illetve a térbeli indexek kezelése miatt.

Az adattárolásra kínálkozó lehetőségek közül mindig az adott alkalmazás igényeit szem előtt tartva célszerű a legelőnyösebb megoldást kiválasztani. Ebben a fejezetben olyan kutatási eredményeimet fogom bemutatni, amelyek hatékonyabbá tehetik a domborzatmodellek tárolását. (1. tézis)

4.1. Piramis index alkalmazása szélsőértékekkel

A piramis index egy jól ismert és széleskörűen alkalmazott eszköz a nagyméretű képek kezelésekor. [47, 53] A lényege az, hogy a képet többféle felbontásban tároljuk, és ezek közül mindig a megjelenítés méretarányának leginkább megfelelőt használjuk. Az egymást követő felbontások általában az eredeti felbontás $1/2$, $1/4$, ... $1/2^n$ részei. A módszer onnan kapta nevét, hogy felfelé haladva az egyes szintek felbontása (és ezáltal az adathalmaz mérete) piramisszerűen egyre kisebb lesz.



4.1.1. ábra. A magasságokra vonatkozó piramis index első szintjének számítása a rácspontok magasságaiból.

Az egyes szintek képeinek egy eleméhez az eggyel nagyobb felbontású kép négy eleme rendelhető. Ezzel a hozzárendeléssel fa gráfok jönnek létre. A viszony az elemek elhelyezkedéséből ered, külön tárolni szükségtelen.

A piramis index egyes képeinek mérete az eredeti kép méretének $1/4$, $1/16$, … $1/2^{2n}$része. A teljes piramis index tárigénye így az eredeti kép méretének harmada. (Kiszámítható a $\sum_{i=1}^{\infty} \frac{1}{2^{2i}}$ alapján.)

A piramis indexet általában nagyméretű képek nagyságrendileg különböző méretarányokban történő megjelenítéséhez használják, ilyenkor ennek megfelelően az indexképek raszterjei az eredeti kép helyileg megfeleltethető raszterjeiben lévő értékek számtani közepét tárolják. Ez a digitális domborzatmodellek esetében egyébként a térfogatszámítások gyorsítására használható fel, mivel az átlagos magasságot a raszter alapterületével megszorozva egyszerűen kapjuk a felületdarab alatti térfogatot. [146]

Figyelmet kell fordítani arra, hogy GRID modellek a magasságokat a rácsháló rácspontjaira vonatkozóan tárolják, a fent bemutatott piramis indexben tárolt magasságok pedig már a kisebb felbontású rácshálók felületeire (négyzeteire) vonatkoznak. Ennek megfelelően az index első szintjét a 4.1.1 ábrán látható módon kell számítani a felületek területére eső kilenc rácspont magasságából kiindulva. A további szinteket már egyszerű átlagolással (az alsó szint négy vonatkozó elemének 0.25-ös súllyal való összegzése) lehet számítani.

Piramis index készíthető oly módon is, hogy az egyes indexképek elemeiben a legkisebb és legnagyobb értékeket, vagyis a raszternek megfeleltethető felületdarab legmagasabb illetve legalacsonyabb pontjának magasságait tároljuk, amivel a domborzatmodell különböző nagyságú felületdarabjainak a befoglaló téglatesteihez jutunk. Egy ilyen téglatest vízszintes kiterjedése az elem elhelyezkedéséből következik, egy négyzettel adható meg, a téglatestek így négyzet alapú hasábok lesznek. A hasáb alsó illetve felső lapjának magassági értelmű elhelyezkedését a tárolt minimum és maximum értékek határozzák meg.

Egy piramis index mérete az eredeti kép méretének harmada, de mivel itt kétféle adatot is tárolunk, a méret az eredeti kép kétharmada lesz. Az index méretének csökkentése érdekében predikciós tömörítés alkalmazható, amelynél várt értéknek az eggyel kisebb

4.1.1. táblázat. Az entrópia értékei a rácssűrűség függvényében a minimális és a maximális értékek esetében

	100	200	400	800	1600	3200
MIN	3,27	3,78	4,23	4,62	4,92	4,85
MAX	3,29	3,87	4,36	4,87	5,37	5,48

felbontású kép helyileg megfeleltethető értékét vesszük. Mivel szélsőértékekről van szó, a négy nagyobb felbontású elem közül legalább egy értékének meg kell egyeznie a várt értékkel, a többi pedig csak kisebb vagy nagyobb lehet nála, attól függően, hogy a maximumokat vagy a minimumokat tároló indexről van-e szó.

A fent bemutatott módszer alkalmazásakor, az egyes indexképek számításánál a kisebb felbontásból a nagyobb felé haladva tudjuk meghatározni az értékeket. Mivel az előszűrések során is ilyen irányban haladunk, elég az indexeknek csak azon értékeit kiszámítani, amelyek az előszűrés során megvizsgált befoglaló idomokkal fedésben vannak.

Az indexeknek a fenti módon történő tömöríthetőségét egy Székesfehérvár környékén kijelölt tesztterületen (575500 < Y < 627000 és 190000 < X < 242000 EOV koordináták által határolt téglalap) próbáltam ki, melyen sík és tagolt domborzatú területek vegyesen találhatóak. A vizsgálatokhoz egy a három másodperces SRTM adatokból levezetett (EOV rendszerbe transzformált) 100 méteres felbontású domborzatmodellt használtam.

A tesztterület 100 méteres felbontású domborzatmodelljéből 200, 400, 800, 1600, 3200 és 6400 méteres felbontású raszter állományokat készítettem, melyek raszterjei az általuk lefedett területdarab legmagasabb illetve legalacsonyabb pontjának magasságait tartalmazták. A következő lépésben megvizsgáltam a 100 - 3200 méteres felbontású állományok értékeit, hogy mennyire térnek el az eggyel kisebb (200 − 6400 méteres) felbontású képnek az adott területre vonatkozó értékeitől. Az így kapott adathalmazoknak, mivel a vizsgált kérdés a tömöríthetőség volt, számítottam az entrópiáját.

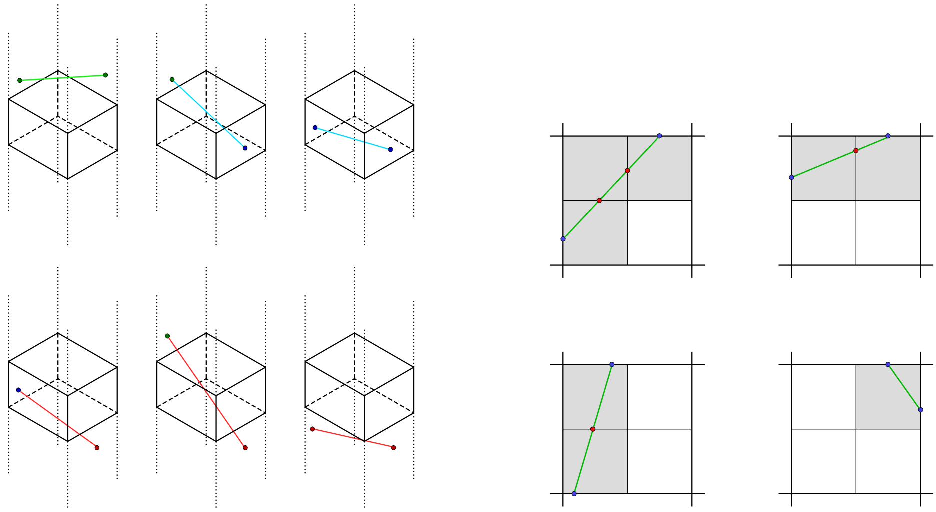
Az entrópiának az informatikában használatos fogalmát Shannon vezette be. A mennyiséget egy X-el jelölt adatforrásra vonatkozóan a következő képlettel számíthatjuk:

$$H(X) = -\sum p_i \log_2 p_i \tag{4.1.1}$$

ahol p_i az egyes események (az adatforrás által közölt adatok) előfordulásának gyakorisága vagy valószínűsége. Az entrópia adott számú lehetséges esemény esetén akkor a legmagasabb, ha minden esemény valószínűsége egyforma, ekkor értéke $\log_2 n$, ahol n a lehetséges események száma. Amennyiben az egyes események valószínűsége eltérő, az entrópia értéke ennél alacsonyabb. Ilyen esetekben a különféle gyakoriságú adatokhoz különféle hosszúságú bitsorozatokat rendelhetünk egy megfelelő kódolás (például Huffmann-kódolás[74]) segítségével, ami biztosítja az adathalmaz tömörítését.

A 4.1.1 táblázat a kapott entrópia értékeket foglalja össze. Jól látható, hogy az entrópia a felbontás csökkenésével növekszik, és hogy ez a növekedés a maximális értékeknél jelentősebb mint a minimális értékeknél. Ezek a jelenségek a domborzat jellemzőiből következnek, hiszen egy nagyobb területen nagyobb magassági eltérések adódhatnak. Az egyes területek legalacsonyabb pontjai völgyekben, legmagasabb pontjai pedig hátak mentén helyezkednek el, melyek közül az előbbiek magassága kevésbé változatos.

Egy alkalmazási lehetőség az előbb bemutatott indexelési eljárásra a láthatósággal



4.1.2. ábra. A szélsőértékes piramis index befoglaló téglatesteinek és a láthatósági elemzések során vizsgált sugár viszonyának lehetséges esetei (bal oldal). A zöld szakaszt (mindkét végpont a téglatest felett) biztosan nem takarja a terep, a piros szakaszokat (legalább egyik végpont a téglatest alatt) pedig biztosan takarja. A kék színű szakaszok (egyik végpont sincs a téglatest alatt, de nincs mindkettő a téglatest felett) esetében a vizsgálatot folytatni kell az index egyel alacsonyabb szintjén a jobb oldalon bemutatott módon.

kapcsolatos vizsgálatoknál [71, 60, 59, 139] adódik. Két pont összeláthatóságának vizsgálatakor azt kell ellenőrizni, hogy a pontok közötti szakasz végig a terep felszíne felett halad-e. Egy térbeli szakaszból számítani tudjuk egy négyzet alapú területre eső rész-szakaszát és ezen szakasz legalacsonyabb pontjának magasságát; a szélsőértékes piramis indexből pedig meg tudjuk mondani, hogy milyen magasságban van a terep legalacsonyabb és legmélyebb pontja egy négyzet alakú területen.

Amennyiben a rész-szakasz legalacsonyabb pontja (ami az alacsonyabban elhelyezkedő végpont lesz) a terület legmagasabb pontja felett helyezkedik el, minden további vizsgálat nélkül megállapíthatjuk, hogy a kérdéses részterületen nincs az összelátást akadályozó felületrész. Ha a rész-szakasz legalacsonyabb pontja a terület legalacsonyabb pontja alatt van, akkor pedig a kitakarás tényét tudjuk egyből megállapítani. Más esetekben rekurzív módon folytatnunk kell a vizsgálatot a területen az eggyel nagyobb felbontású szélsőértékes piramis index segítségével. (4.1.2 ábra.)

A fenti módszer nélkül a láthatóság vizsgálatához a két vizsgált pont közötti távolsággal (t) arányos számú műveletre van szükségünk. A szélsőértékes piramisindex alkalmazásával ez akár a távolság logaritmusával arányos lépésre is csökkenthető. A műveletek pontos száma a konkrét domborzati viszonyoktól is függ, de életszerű esetekben az $O(\log t)$-hez közelít az $O(t)$ helyett.

A szélsőértékes piramis index elve sok tekintetben hasonlít az egyik legelső hazai digitális domborzatmodellre a DTM-200 adatbázisra. Ebben a hetvenes években a PKI[1] Mikrohullámú és Űrtávközlési Osztályán létrehozott modellben 200×200 méteres te-

[1]Posta Kísérleti Intézet, bővebben róla a http://itf2.njszt.hu/intezmeny/pki címen található anyagban lehet olvasni.

rületekre vonatkozóan tárolták a terepfelszín kérdéses darabjának legnagyobb magasságát, valamint a területen belül előforduló legnagyobb magasságkülönbséget, így információtartalma hasonló volt az előzőekben bemutatott piramis index egy szintjéhez; bár a magasságkülönbségre vonatkozó adat csak kategorizálva, 5 méteres felbontással volt ismert, a 40 méter feletti magasságkülönbségek pedig egy közös kategóriába kerültek. A DTM-200 a mikrohullámú távközlési hálózatok tervezésének támogatására készült, ezért a domborzati adatok mellett a terep fedettségére vonatkozó adatokat is tartalmazott.[2]

4.2. A 2+1 dimenziós R-fa alkalmazása TIN modellek tárolásakor

Az R-fa (R-tree) index egy vektoros térinformatikai adatok térbeli feltételek alapján történő gyors előszűrésére kidolgozott térbeli index. Lényege, hogy egy olyan fa típusú gráfot hoz létre, amelynek csomópontjaihoz téglalapokat (3D adattárolás esetén téglatesteket) rendel úgy, hogy az egyes csomópontok téglalapjai teljes egészében tartalmazzák a belőlük származó csomópontok téglalapjait, a fa levelein pedig az indexelendő objektumok befoglaló téglalapjai helyezkednek el. A fa gyökeréhez tartozó téglalap így valamennyi indexelt objektum befoglaló téglalapját tartalmazza. A módszerrel kapcsolatban az első publikáció [68] óta számos cikk született, sokféle változatát dolgozták ki.

Az R-fa index alkalmas a TIN típusú domborzatmodellek háromszögeinek indexelésére is. A tetszőleges dimenziószám esetén alkalmazható indexelési módszernek ebben az esetben a három dimenziós változatát lenne kézenfekvő használni, de a domborzat és a modellezésére használt TIN háló több olyan tulajdonsággal is rendelkezik, amely ennek a döntésnek az átgondolására késztet.

Egy domborzatmodell által leírt felület kiterjedése általában több nagyságrenddel nagyobb vízszintes, mint magassági értelemben. További fontos tulajdonság, hogy a háromszögháló elemeinek a vízszintes vetületei hézag- és átfedésmentesen fedik le a síkot. A magassági adatok ennek ellenére nagyon fontos információt hordoznak.

Az R-fa index kialakításának és kezelésének egy lényeges metódusa a csomópontok szétvágása. Erre akkor van szükség, ha a csomópontból induló élek száma egy új él beszúrását követően meghaladná a maximálisan tárolható élek számát. Ilyenkor a csomópont elemeit szétválogatjuk két, egymástól a térben lehetőleg minél jobban elkülönülő csoportra, és az így létrejövő kettő új csomópontot a régit lecserélve bejegyezzük abba a csomópontba, ahonnan az származott. Ha ezáltal abban a csomópontban is betelik a hely, a vágást rekurzívan folytatjuk, felfelé haladva a fában; szükség esetén a gyökeret is kettévágjuk és egy új, eggyel magasabb szinten elhelyezkedő gyökérbe jegyezzük be az így kapott részeket.

A csomópontok szétvágására többféle algoritmus létezik. Már a [68] is több módszert adott meg, majd később mások további algoritmusokat is publikáltak. A csomópontok vágására használt algoritmus kiválasztása nagyban befolyásolja az indexelés hatékonyságát a létrejövő index-struktúra tekintetében, illetve a csomópont vágására fordítandó, és ezáltal a beszúrásokhoz és módosításokhoz szükséges idő kérdésében.

[2] A DTM-200-ról bővebben a http://itf.njszt.hu/23r4r23r/uploads/2015/06/tiszoczi_gallyas.ppt címről letölthető előadásban lehet olvasni.

A teljeskörű keresés (Exhaustive Search) megvizsgál minden lehetséges felosztást, és ezek közül kiválasztja azt, amelyiknél a létrejövő két új csomóponthoz tartozó befoglaló téglalapok területeinek összege minimális. Ezzel a módszerrel hatékony index alakítható ki, viszont az időigénye nagy, hiszen a csomópontokban elférő befoglaló téglalapok számával exponenciálisan növekszik, mivel n darab elemet $2^{n-1} - 1$ féle módon lehet két csoportba osztani úgy, mindegyik létrehozott csoportba legalább egy elem kerüljön, és ezeket a lehetőségeket a teljeskörű keresésnél mind meg kell vizsgálni.

A négyzetes metódus (Quadratic Method) és a lineáris metódus (Linear Method) egymáshoz hasonló elven működnek: kiválasztanak két elemet a két csoport kezdőelemének (`PickSeed`), majd a többi elemet egymás után megvizsgálva ezen csoportok valamelyikéhez rendelik (`PickNext`). A különbség a két módszer között abban rejlik, hogy a `PickSeed` és a `PickNext` műveletek a négyzetes metódus esetében úgy működnek, hogy az elemek számával négyzetesen arányos számú lépéssel lefutó algoritmust eredményeznek, a lineáris metódus esetében pedig úgy, hogy a csomópont vágásához csak az elemek számával egyenesen arányos számú lépésre van szükség. Az indexelt tér dimenziójának számával a számítási igény minden bemutatott módszer esetében lineárisan nő.

Meg kell még említeni az R*-fa (R*-tree) indexet, ami az R-fa index egy módosított változata [34]. Ez abban tér el az eredeti megoldástól, hogy az új elemek beszúrásakor más módon jár el az optimálisabb keresőfa létrehozása érdekében. A nagyobb hatékonyság ára az, hogy összetettebb algoritmusokat használ. A keresés és a törlés tekintetében az R-fával azonos elven működik.

A TIN típusú digitális domborzatmodellek háromszögeinek R-fa index segítségével történő tárolása esetén lehetőség van arra, hogy a fa kialakításakor, tehát a csomópont vágások végrehajtásakor, csak a befoglaló idomok vízszintes helyzetét vegyük figyelembe, viszont a befoglaló idomok adatai között már a magassági információkat, vagyis a téglatest alsó és felső lapjának magasságait is tároljuk.

Az így kapott 2+1 dimenziós R-fa segítségével elvégezhető minden olyan művelet, ami egy háromdimenziós R-fával, viszont annál gyorsabban kezelhető, mert a csomópont vágásokat csak két dimenzióban kell elvégezni, és amint azt láttuk, ennek a műveletnek a számításigénye a dimenziószámmal együtt növekszik. Ezen kívül az index is optimálisabb felépítésű lesz, hiszen ha a csomópontok vágását három dimenzióban végeznénk, akkor az a befoglaló téglatestek térfogatának minimalizálására törekedne, ami eltérő eredményt ad attól, mint amikor az előzőek vízszintes vetületeként előálló téglalapok területét minimalizáljuk.

A módszer arra támaszkodik, hogy egy domborzatmodell kiterjedése vízszintes értelemben általában sokkal nagyobb, mint magassági értelemben. Más esetekben, például egy épület modelljénél már nyilván nem lenne hatékonyan használható.

5. fejezet

Lejtésviszonyok eloszlásának ábrázolása

A terep lejtésviszonyai, amit a gyakorlatban általában a kitettséggel és a lejtőkategóriával jellemeznek, nagyon fontosak a terület hasznosíthatósága szempontjából. Az esésvonal irányára és a terep esésére vonatkozó információk sok esetben együtt mutatják meg, hogy alkalmas-e a terület valamilyen célra, például érdemes-e oda szőlő- vagy gyümölcsültetvényt telepíteni. Előfordulhat az is, hogy egy nagyobb terület esetében ki szeretnénk mutatni, hogy a különféle lejtésviszonyú területek milyen eloszlásban vannak ott jelen.

Ebben a fejezetben bemutatom azokat a domborzatmodellekkel kapcsolatos eljárásokat amelyeket a lejtésviszonyok eloszlásának ábrázolására dolgoztam ki. (2. tézis)

5.1. Ábrázolási lehetőségek

A felmerülő feladat egyszerűen megoldható egy táblázat segítségével, aminek soraiban a lejtőkategóriák, oszlopaiban pedig a kitettségek szerepelnek. A táblázat celláiban az adott kitettségű és lejtőkategóriájú részek összterülete vagy százalékos aránya szerepelhet. A sík területek sorában a különféle kitettségekhez tartozó cellákat akár össze is lehet vonni, és az így keletkező cellában lehet elhelyezni a síknak tekintett területek adatait. Szükség esetén a szokásos (a 2.4.1 és a 2.4.3 táblázatokban használt) besorolásoknál részletesebb felosztáson alapuló táblázatot is lehet készíteni, mint amilyen a 5.1.1 táblázatban is látható.

Egy ilyen táblázatnak az adatai grafikusan is ábrázolhatóak. Ennek az egyik legegyszerűbb, kézenfekvő módja lenne, ha egy térhatású oszlopdiagram segítségével tennénk összevethetővé a táblázat celláiban található számokat. (5.2.2 ábra bal oldala) Ez a módszer azonban nem túl látványos, és nagy gyengesége még az is, hogy a különféle kitettségek (vagy egyéb az esésvonal iránya szerint felvett osztályok) között azonos távolság lenne minden lejtőkategóriában, pedig annak jelentősége meredekebb terep esetében sokkal fontosabb mint egy közel sík területen. További problémát jelent, hogy az ábrán két ellentétes szélére kerülő oszlopsorok valójában szomszédosak egymással.

5.1.1. táblázat. Egy mintaterületen belüli különféle lejtésviszonyú területek nagysága hektárban lejtés (oszlopok) és kitettség (sorok) szerint táblázatba foglalva.

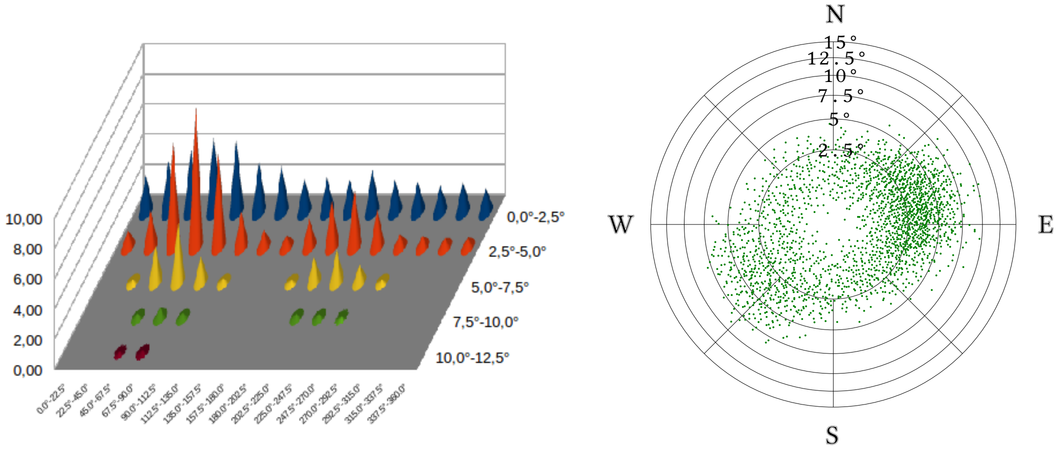
	0.0°-2.5°	2.5°-5.0°	5.0°-7.5°	7.5°-10.0°	10.0°-12.5°	ÖSSZ
0.0°-22.5°	2.45	1.10	0.00	0.00	0.00	3.55
22.5°-45.0°	3.40	2.45	0.15	0.00	0.00	6.00
45.0°-67.5°	4.15	6.95	2.40	0.25	0.05	13.80
67.5°-90.0°	5.20	9.40	3.95	0.65	0.10	19.30
90.0°-112.5°	4.75	6.25	1.75	0.35	0.00	13.10
112.5°-135.0°	3.30	2.35	0.25	0.00	0.00	5.90
135.0°-157.5°	3.10	1.15	0.00	0.00	0.00	4.25
157.5°-180.0°	2.25	0.80	0.00	0.00	0.00	3.05
180.0°-202.5°	2.30	1.90	0.15	0.00	0.00	4.35
202.5°-225.0°	2.15	3.05	1.65	0.25	0.00	7.10
225.0°-247.5°	2.00	3.80	2.15	0.20	0.00	9.00
247.5°-270.0°	1.75	2.55	1.15	0.05	0.00	5.90
270.0°-292.5°	2.05	0.85	0.15	0.00	0.00	3.00
292.5°-315.0°	1.75	0.75	0.00	0.00	0.00	2.50
315.0°-337.5°	2.05	0.60	0.00	0.00	0.00	2.65
337.5°-360.0°	1.60	0.65	0.00	0.00	0.00	2.25
ÖSSZ	45.45	44.60	13.75	1.75	0.15	105.70

5.2. Lejtésviszonyok eloszlását ábrázoló diagram

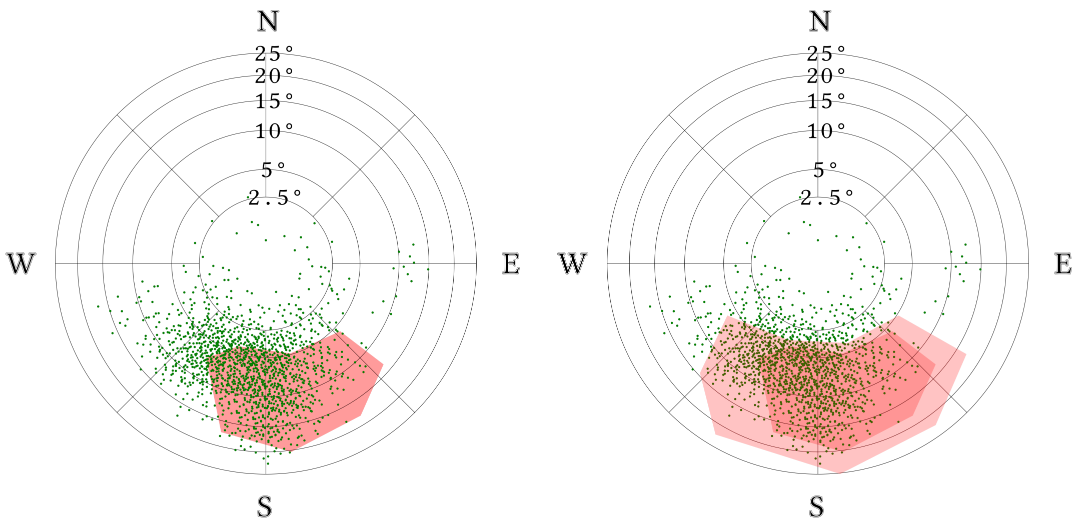
A megoldást egy poláris koordináta-rendszert alkalmazó diagram használata jelentheti. Ebben az egyes lejtőkategóriáknak gyűrűk felelnek meg, a sík területeket a diagram közepén elhelyezkedő kör jelképezi. A gyűrűket sugárirányú vonalakkal a kitettségeknek megfelelő szektorokra felvágva kapjuk azokat a grafikus elemeket, amelyek a diagram közepén lévő (a sík területekhez rendelt) körrel együtt a terep adott lejtőkategóriájú és kitettségű részeit jelképezik. Ezekkel a grafikus elemekkel kell kifejezni, hogy a vizsgált területen mekkora a lejtésviszonyai alapján az adott kategóriába sorolt felületrész. Erre a célra a diagram kérdéses elemének (egy szektor vagy a sík területeket jelképező kör) felületére a terep adott jellegű részeinek összterületével azonos számú pontot szórunk szét véletlenszerűen vagy más véletlen-jellegű módon, például egy Halton-sorozattal. Ezzel a módszerrel az eltérő irányú (kitettségű) lejtőket jelképező grafikus elemek annál távolabb kerülnek egymástól, minél meredekebbek; a közel sík területek pedig egymás közlében vannak még jelentősebben eltérő irány esetében is. Természetesen lehetőségünk van a szokásos kategorizálásokból eredőnél több osztály létrehozására, ezáltal a diagram „felbontása” finomítható.

A diagram koordináta-rendszerében körül tudjuk határolni azokat a részeket amelyek valamilyen szempontból (például valamilyen növény termesztése) megfelelőnek bizonyulnak, így a diagramra tekintve már azt is látjuk, hogy a vizsgált terület lejtésviszonyainak eloszlása hogyan viszonyul az ideálisnak tekintetthez. (5.2.2ábra) A körülhatárolás során egyetlen éles halmaz helyett akár több kategóriát vagy átmeneteket is alkalmazhatunk.

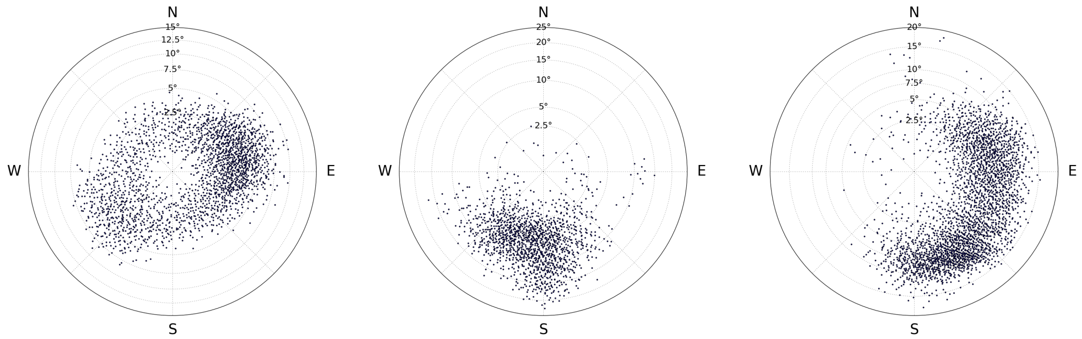
A diagram koordinátarendszerében körülhatárolt területeket a térképen is meg lehet



5.2.1. ábra. A domborzati viszonyok eloszlásának ábrázolására egy klasszikus diagrammal (bal oldal) és a javasolt módszerrel (jobb oldal). A klasszikus diagramon hengeres oszlopok helyett kúpokat használtam a kitakarások minimalizálása érdekében. A javasolt diagramon minden egyes zöld színű pont egy egységnyi területű felületdarabot jelent, aminek a lejtésviszonyai a diagram poláris koordinátarendszeréből olvashatóak le.



5.2.2. ábra. Ideálisnak tekintett lejtésviszonyok elhatárolása a diagram koordinátarendszerében egyetlen éles halmazzal (bal oldali diagram) és több kategóriával (jobb oldali diagram).



5.3.1. ábra. A Matplotlib segítségével készített diagramok. (A többi ábra lejtésviszonyok eloszlását bemutató diagramjai az SVG állományokat előállító megoldással készültek.)

jeleníteni egy domborzatmodell alapján. Ez alapesetben egy logikai értékeket tartalmazó raszter réteget eredményez annak megfelelően, hogy a vizsgált területdarabon a lejtő iránya és nagysága a diagramon lehatárolt részre esik-e. A térképi megjelenítés átmeneti kategóriák vagy folyamatos átmenet esetén is alkalmazható, csupán a létrejövő raszter réteg típusa lesz egész vagy lebegőpontos szám.

5.3. A diagramok előállítása

A bemutatott diagramok előállításához Python nyelven készítettem programokat. A diagramhoz szükséges adatokat kigyűjtő program az OSGeo Python könyvtárának[1] GDAL [140] modulja segítségével segítségével olvasta be a domborzatmodell adatait egy NumPy [138] tömbbe. A terület poligonjának leírását egy WKT formátumú szövegként kapta meg a program.

Azokon a helyeken, ahol a rácsháló egy négyzetének középpontja a vizsgált területre esett, a program kiszámította a kérdéses felületdarab lejtésére vonatkozó adatokat, majd kiírta azokat egy szöveges állományba.

Ebből a szöveges állományból dolgozott a diagramokat előállító program, ami a kapott adatok alapján egy SVG állományt hozott létre. A létrehozott SVG állomány tartalmazott minden a diagramhoz kapcsolódó grafikus elemet, és az adatok alapján felkerültek rá a lejtésviszonyok eloszlását szemléltető pontok is.

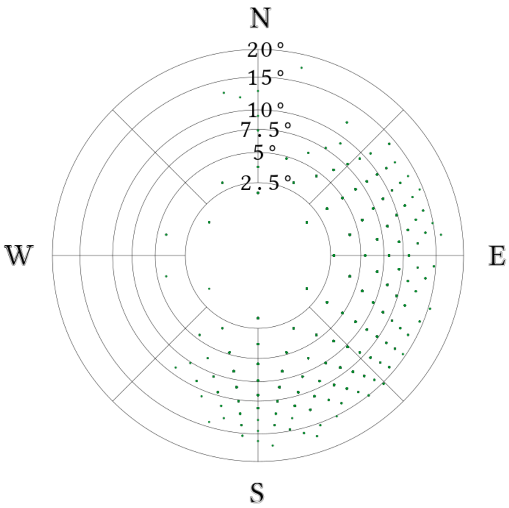
Később a jól testreszabható diagramok készítésére megalkotott Matplotlib[2] [75] Python modul segítségével működő programot is készítettem. (5.3.1. ábra) Itt nagy segítségemre volt, hogy a poláris koordinátarendszer alkalmazása és a pontokból álló, eloszlást kifejező diagramok (scatter) a Matplotlib-ben alapvető eszközök, így ez a program jóval egyszerűbb és tömörebb volt, mint az előző [6].

5.4. A megjelenítés részletkérdései

Az elemzések során SRTM domborzatmodelleket használtam, amelyek a magasságot méteres élességgel tartalmazták. Mivel a rácsháló négyzeteinek csúcsaihoz tartozóan meg-

[1] bővebben lásd: https://wiki.osgeo.org/wiki/OSGeo_Python_Library

[2] A Matplotlib a https://matplotlib.org/ oldalról tölthető le.



5.4.1. ábra. A méter élességű magasságokból közvetlenül generált diagram (bal), és a magasságok minimális véletlenszerű módosításával kapott eredmény (jobb).

adott magasságok különbségei így viszonylag kevés számú esetet eredményeznek, a diagram pontjai egy pozícióba esve nem fejezhetnék ki jól a lejtésviszonyok eloszlását, mivel a diagramot szemlélők nem tudnák eldönteni, hogy hány pont van egymáson az egyes helyeken. (5.4.1.ábra) A probléma elkerülése érdekében a rácspontok magasságaihoz egy $-0,5$ és $+0,5$ közötti egyenletes eloszlású véletlen számot adtam hozza a lejtési adatok számítása előtt.

Fontos a pontok méretének (vagyis a pontokat megjelenítő körök átmérőjének) megfelelő megválasztása. Ha az egymással átfedő pontok teljesen (vagy közel teljesen) kitöltik a síkot, akkor a különféle sűrűségű részek vizuális elkülönítése nehézzé válik.

6. fejezet

Terepszerkezeti formák elkülönítése

A terep felszínének pontjait különféle kategóriákba sorolhatjuk azok jellege szerint. A topográfiában szokásos megnevezéseket használva csúcsoknak (kúp) illetve mélypontnak (teknő) hívjuk a terepfelszín lokális maximumait illetve minimumait. A hátvonalak (gerincvonalak) és a völgyvonalak a hegyhát legmagasabb illetve a völgy legalacsonyabb pontjait összekötő vonalak. A domborzat egyes részein nyergek alakulnak ki, ahol több gerincvonal illetve völgyvonal találkozik. A terepnek azokat a pontjait, ahol semmiféle az előbbiekben bemutatott jellegzetes pont vagy idomvonal nem található, lejtőnek vagy (ha a terep lejtése nem jelentős a vizsgált helyen) sík(ság)nak nevezzük.

A topográfiában megkülönböztetnek még további ún. mellékidomokat is, de a következőkben bemutatott elemzések szempontjából ezek nem különülnek el az előbbiekben bemutatottaktól, illetve azokból felépíthetőek.

Ebben a fejezetben bemutatom a terepszerkezeti jellemzők leírására kidolgozott, az irányszög szerinti magasságkülönbségekre illesztett Fourier-sor paraméterein alapuló módszeremet; illetve azt, hogy ezeknek a elemzéseknek az eredményéből kiindulva hogyan nyílik lehetőség a terepszerkezeti formák fuzzy alapú elkülönítésére. (3. tézis)

Fontos megjegyezni, hogy a következőekben tárgyalt terepszerkezeti formák között felületszerű (síkságok, lejtők), vonalas (gerinc- és völgyvonalak) és pontszerű (csúcsok, mélypontok, nyeregpontok) elemek is vannak. A bemutatott eljárások ezzel szemben a terepfelszín elemi darabjait sorolják be a fentieknek megfelelő osztályokba. Ilyenkor nyilvánvalóan a kérdéses idomvonalak és pontok közélében található területek sorolódnak a megfelelő kategóriákba.

6.1. Terepszerkezeti formák felismerésének klasszikus módszerei

A terep különféle jellegű pontjainak besorolására többféle módszer létezik. A következőkben a [116] alapján két közismert eljárást is be fogok mutatni, mielőtt javaslatot tennék további módszerekre.

Egy megfelelően megválasztott r sugarú kör mentén 15 fokonként kiszámítjuk a terep magasságát, majd képezzük az egymással ellentétes oldalon lévő pontokat összekötő szakaszok felezőpontjának és a vizsgált pontnak a magasságkülönbségeit. Ezt követően megszámoljuk, hogy ezek között a magasságkülönbségek között mennyi olyan pozitív

(N^+) illetve negatív (N^-) érték van, aminek az abszolút értéke meghalad egy meghatározott érdességi tényezőt (E). Ha N^+ és N^- értéke is nulla (vagyis minden magasságkülönbség az E érdességi tényező értékén belül volt), akkor a pontot sík területnek tekintjük. Ha az összes magasságkülönbség pozitív vagy negatív volt, akkor a pont mélypontnak vagy kúpnak minősül. Ha $N^+ = 0$ és $N^- > 0$, akkor a pont gerincvonalra, a fordított esetben ($N^- = 0$ és $N^+ > 0$) pedig völgyvonalra esik. A többi (az előbbi szabályok szerint még nem besorolt) pontot lejtőnek tekintjük.

A módszer alkalmazásakor fontos az r sugár és az E érdességi tényező helyes megválasztása. Kisebb sugár mellett a szerkezeti vonalak megszakadhatnak, nagyobb sugár esetén pedig sávvá szélesedhetnek. A módszer a nyeregpontokat nem sorolja külön kategóriába.

Egy másik módszert alkalmazva a vizsgált ponttal azonos középpontú r sugarú kör mentén haladva képezni kell a kerületi pontok és vizsgált pont magasságkülönbségeit. Az így kapott számsorból számítani kell a pozitív (S^+) és a negatív (S^-) elemek összegét, meg kell számlálni a negatív elemek számát (L) valamint azt, hogy a kör mentén haladva az érték hányszor változtatja meg az előjelét (N).

Ha a pont magasabban van valamennyi a környezetében található ponttól, akkor kúppont, ha alacsonyabban, akkor mélypont. Amennyiben S^+ és S^- abszolút értékeinek összege nem lép túl egy E^+ küszöbértéket, a területet síknak tekintjük. Ha a pont környezetében a pozitív értékek jellemzőek akkor völgyvonalra, ha a negatívak, akkor gerincvonalra esik. Amennyiben a pozitív és a negatív értékek száma azonos, akkor az előjelváltások számát (N) kell megvizsgálni. Amennyiben $N = 2$, a pont lejtőnek tekinthető, ilyenkor a magasságkülönbségek görbéje periodikusan egy hullámot ír le. Ha $N = 4$, akkor a pont nyeregpont, az előbbi görbe ilyenkor két hullámot is leír egy periódus alatt.

6.2. Terepszerkezeti formák jellemzése irányszög szerinti Fourier-sorokkal

Az előző módszerekben meghatározott mennyiségek helyett az r sugarú kör mentén a ponthoz képest meghatározott magasságkülönbségekre egy másodfokú Fourier-sort is felírhatunk az irányszög (δ) függvényében, és ennek paramétereiből is megpróbálhatjuk eldönteni a pont jellegét. A sor paramétereiből ugyanis minden az előző módszerekben a pont besorolásához használt jellemző kiolvasható.

Egy $f(x)$ függvényt a következő módon közelíthetünk egy N-ed fokú Fourier-sorral:

$$f(x) \sim s_N(x) = \frac{a_0}{2} + \sum_{i=1}^{N} \left(a_i \cos(ix) + b_i \sin(ix)\right) \tag{6.2.1}$$

ahol az összesen $2N + 1$ darab a_i-vel és b_i-vel jelölt együtthatók értékeit a következő módon számíthatjuk:

$$\begin{aligned} a_i &= \tfrac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(ix)\, dx \\ b_i &= \tfrac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(ix)\, dx \end{aligned} \tag{6.2.2}$$

Ezek a közelítések a felhasznált trigonometriai függvények jellegéből adódóan egy 2π periódusú függvényre vagy valamilyen más függvénynek egy ilyen hosszúságú szakaszára alkalmazhatóak.

A fenti közelítő elvet alkalmazhatjuk a vizsgált ponttól r távolságban található, egymástól δ irányszögük alapján megkülönböztethető pontoknak a vizsgált ponthoz viszonyított magasságkülönbségére is. A Fourier-sor együtthatói ekkor a következő módon számíthatóak egy x, y koordinátákkal megadható pont r sugarú környezetére:

$$\begin{aligned} a_i &= \tfrac{1}{\pi} \int_{0^\circ}^{360^\circ} \left(h\left(x + r\sin\delta, y + r\cos\delta\right) - h\left(x, y\right)\right)\cos\left(i\delta\right) d\delta \\ b_i &= \tfrac{1}{\pi} \int_{0^\circ}^{360^\circ} \left(h\left(x + r\sin\delta, y + r\cos\delta\right) - h\left(x, y\right)\right)\sin\left(i\delta\right) d\delta \end{aligned} \tag{6.2.3}$$

A $h\left(x, y\right)$ a kétváltozós függvénynek tekintett terepfelszín. A 2π, vagyis 360 fok hosszúságú periódus az irányszögből értelemszerűen adódik. A gyakorlatban az integrálás helyett M számú pontban számított értékek összegzését végezzük. Az együtthatók számítása így a következő összefüggésekkel történik:

$$\begin{aligned} a_i &= \tfrac{2}{M} \sum_{j=0}^{M-1} \left(\left(h\left(x + r\sin\tfrac{2\pi j}{M}, y + r\cos\tfrac{2\pi j}{M}\right) - h\left(x, y\right)\right)\cos\left(i\tfrac{2\pi j}{M}\right)\right) \\ b_i &= \tfrac{2}{M} \sum_{j=0}^{M-1} \left(\left(h\left(x + r\sin\tfrac{2\pi j}{M}, y + r\cos\tfrac{2\pi j}{M}\right) - h\left(x, y\right)\right)\sin\left(i\tfrac{2\pi j}{M}\right)\right) \end{aligned} \tag{6.2.4}$$

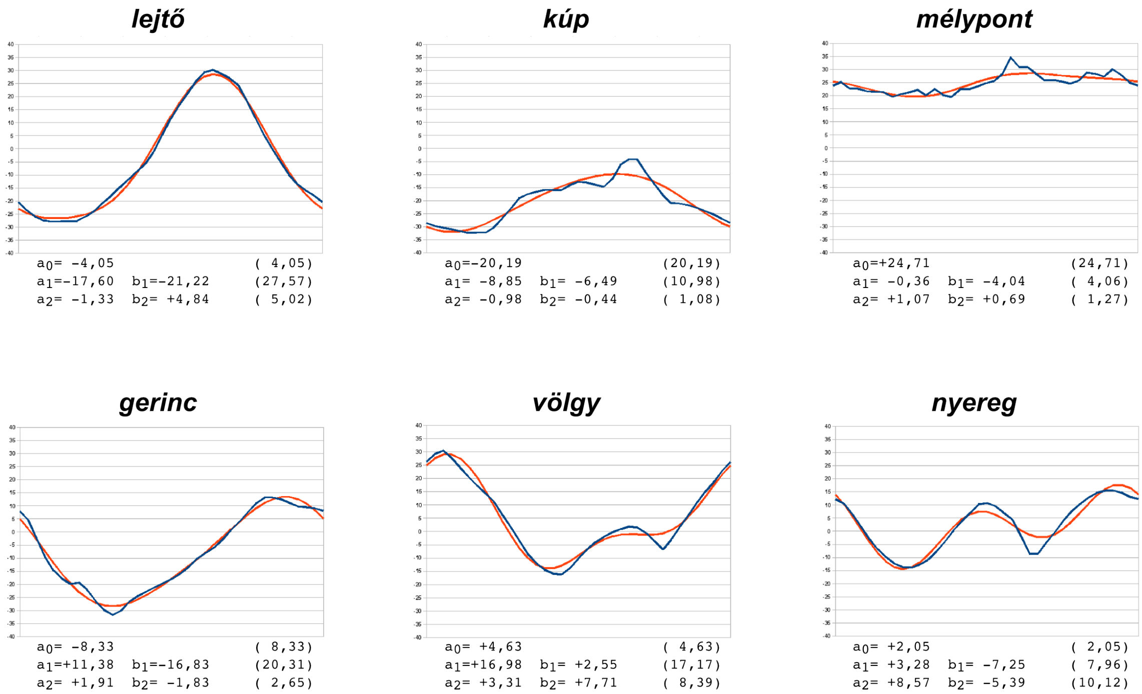
A fentiek alkalmazása során szükséges a $h\left(x, y\right)$ ismerete, ami az esetünkben azt jelenti, hogy a rendelkezésre álló domborzatmodell alapján tetszőleges vízszintes pozícióban meg kell tudnunk állapítani a magasságot. Mivel ezek a pozíciók a vizsgált pont körül egy kör mentén helyezkednek el, csak nagyon ritkán esnek a domborzatmodell támpontjaira, így valamilyen interpolációt is alkalmaznunk kell.

Az a_0 együttható értéke azt fejezi ki, hogy a pont mennyivel van átlagosan magasabban vagy alacsonyabban az r sugarú környezeténél. Az a_1 és b_1 együtthatók értékeiből a felület dőlésére lehet következtetni. Ha az a_2 és $b_2$együtthatók értékei jelentősek, akkor abból arra következtethetünk, hogy a pont egy nyeregpont vagy (ha közben a_1 és b_1 értékek is jelentősek) egy idomvonalon található. (6.2.1. ábra)

Az r sugár értéke különféle lehet, ami különböző $a_i$és $b_i$értékeket eredményezhet, amelyek így r függvényében értelmezhetőek. Lehetőségünk van az a_i és a $b_i$értékeket tetszőleges számú és értékű r mellett kiszámítani, majd az eredményekre egy N-ed fokú polinomot illeszteni, ami az $y = c_0 + c_1 x + c_2 x^2 + \cdots + c_N x^N$ módon $N + 1$ adattal adható meg. Egy M-ed fokú Fourier-sor paramétereinek N-ed fokú polinommal való kifejezése így összesen $\left(2M + 1\right)\left(N + 1\right)$ adatot jelet. Ezek az adatok megadják a domborzat közelítő leírását egy pont környezetében.

A vizsgálatot fordítva is el lehet végezni. Egy pontból kiindulva különféle irányokban egy-egy N-ed fokú polinommal írjuk le a felület függőleges metszeteit, majd ezeknek a polinomoknak a paramétereire írunk fel Fourier-sorokat a metszet irányszögének függvényében. Ennek a megoldásnak az a hátránya, hogy a polinomok az $r = 0$ esetben az irányszög függvényében különböző értékeket vehetnek fel, így a felület nem lesz folytonos, ezért részletesebben nem is foglalkoztam ezzel a módszerrel.

Az a_i és b_i paraméterek előállításának egyik kézenfekvő módja a (6.2.4) egyenletben



6.2.1. ábra. Különféle terepszerkezeti formákhoz tartozó pontok környezetének vizsgálata a hozzájuk tartozó másodfokú Fourier-sor segítségével. A kék vonal a terepfelszínnek a pont környezetében vett metszetét mutatja 10 fokos mintavételezési sűrűséggel. A piros vonal az ugyanezekre az adatokra illesztett másodfokú Fourier-sor képe. A diagramok alatt a Fourier sor paraméterei is megtalálhatóak. A zárójelben az $|a_0|$, a $\sqrt{a_1^2+b_1^2}$ illetve a $\sqrt{a_2^2+b_2^2}$ értékek vannak.

bemutatott numerikus integrálok számítása, amihez a terepfelszín magasságát (a képletekben $h\left(x,y\right)$-al jelölve) a vizsgált pont köré rajzot kör mentén M darab pontban kell meghatározni valamilyen interpolációs módszerrel, majd ezeket az értékeket (pontosabban a vizsgált pont magasságával képzett különbségeiket) szorozni kell az irányszög meghatározott trigonometriai függvényével, és az így kapott szorzatok összegét kell számítani. A magasságok meghatározását végezhetjük olyan interpolációs módszerrel, amelyik egy pont magasságát a környező rácspontok súlyozott átlagaként adja meg, mint például a (**??**) összefüggéssel leírható bilineáris interpoláció.

A $h\left(x,y\right)$ függvény kiszámítandó értékeit ezzel a rácspontok magasságainak lineáris kombinációjára vezethetjük vissza, amelyeket azután az irányszög megfelelő, a kérdéses pontra nézve konstans értékű trigonometriai függvényével is szorozhatunk, majd összegezhetünk is. A végeredmények a 6.2.2. ábrán bemutatottakhoz hasonló konvolúciós szűrők lesznek a (rácstávolsághoz viszonyított) sugártól és az interpolációs módszertől függően. Ezekkel a konvolúciós szűrőkkel már a legtöbb térinformatikai szoftverben lehetőségünk van arra, hogy az a_i és b_i paramétereket egy vizsgált domborzatmodell minden rácspontjára vonatkozóan gyorsan és egyszerűen ki tudjuk számítani.

Minden paraméterhez másik konvolúciós szűrő tartozik. Az eredményként kapott raszteres állomány a kérdéses paraméter területi változását fogja ábrázolni. Ha a paraméterekre támaszkodva akarunk további jellemzőket levezetni, akkor aritmetikai műveleteket kell végeznünk a konvolúciós szűrővel kapott állományok között.

Az a_0, a_1, b_1, a_2 és b_2 adatokból és egy újonnan bevezetett e érdességi tényezőből számíthatjuk a következő paramétereket:

$$\begin{aligned}
P_0 &= 1-\tfrac{a_0^2+a_1^2+b_1^2+a_2^2+b_2^2}{e^2+a_0^2+a_1^2+b_1^2+a_2^2+b_2^2}\\
P_1 &= \tfrac{a_1^2+b_1^2-a_0^2-a_2^2-b_2^2}{e^2+a_0^2+a_1^2+b_1^2+a_2^2+b_2^2}\\
P_2 &= \tfrac{a_0\sqrt{a_2^2+b_2^2}}{e^2+a_0^2+a_1^2+b_1^2+a_2^2+b_2^2}\\
P_3 &= \tfrac{a_0^2-a_1^2-b_1^2-a_2^2-b_2^2}{e^2+a_0^2+a_1^2+b_1^2+a_2^2+b_2^2}\\
P_4 &= \tfrac{a_2^2+b_2^2-a_0^2-a_1^2-b_1^2}{e^2+a_0^2+a_1^2+b_1^2+a_2^2+b_2^2}
\end{aligned} \tag{6.2.5}$$

Az e értéke méterben (vagy más a magasság mérésére használt mértékegységben) értendő. Mivel a felületmodellből levezetett a_i és b_i paraméterek mértékegysége szintén hasonló, a P_i paraméterek dimenzió nélküli számok lesznek. A P_0értéke 0 és 1, a többi fent bevezetett paraméter értéke pedig -1 és 1 között változik.

Minél nagyobb a P_0értéke (minél inkább közelít 1-hez) a terep a vizsgált pont környezetében annál inkább síknak tekinthető; a P_1 értéke pedig a lejtős területeken nagy. A P_2 abszolút értéke az idomvonalak mentén lesz jelentős; a gerincvonalak mentén a -1-hez, a völgyvonalak mentén pedig az 1-hez közelít. A P_3 értéke a terepfelszín lokális magassági szélsőértékeinek közelében lesz jelentős; ha az $a_0 < 0$ akkor ezeken a helyeken csúcsok, ha pedig $a_0 > 0$, akkor pedig teknők vannak. A P_4 értéke a nyeregpontok környezetében lesz magas.

+0,00000	+0,00102	+0,01786	+0,03005	+0,01786	+0,00102	+0,00000
+0,00102	+0,04211	+0,05198	+0,03451	+0,05198	+0,04211	+0,00102
+0,01786	+0,05198	+0,00162	+0,00000	+0,00162	+0,05198	+0,01786
+0,03005	+0,03451	+0,00000	-1,00000	+0,00000	+0,03451	+0,03005
+0,01786	+0,05198	+0,00162	+0,00000	+0,00162	+0,05198	+0,01786
+0,00102	+0,04211	+0,05198	+0,03451	+0,05198	+0,04211	+0,00102
+0,00000	+0,00102	+0,01786	+0,03005	+0,01786	+0,00102	+0,00000

+0,00000	-0,00051	-0,00554	+0,00000	+0,00554	+0,00051	+0,00000
-0,00088	-0,02943	-0,02295	+0,00000	+0,02295	+0,02943	+0,00088
-0,01681	-0,04571	-0,00114	+0,00000	+0,00114	+0,04571	+0,01681
-0,02967	-0,03399	+0,00000	+0,00000	+0,00000	+0,03399	+0,02967
-0,01681	-0,04571	-0,00114	+0,00000	+0,00114	+0,04571	+0,01681
-0,00088	-0,02943	-0,02295	+0,00000	+0,02295	+0,02943	+0,00088
+0,00000	-0,00051	-0,00554	+0,00000	+0,00554	+0,00051	+0,00000

+0,00000	-0,00088	-0,01681	-0,02967	-0,01681	-0,00088	+0,00000
-0,00051	-0,02943	-0,04571	-0,03399	-0,04571	-0,02943	-0,00051
-0,00554	-0,02295	-0,00114	+0,00000	-0,00114	-0,02295	-0,00554
+0,00000	+0,00000	+0,00000	+0,00000	+0,00000	+0,00000	+0,00000
+0,00554	+0,02295	+0,00114	+0,00000	+0,00114	+0,02295	+0,00554
+0,00051	+0,02943	+0,04571	+0,03399	+0,04571	+0,02943	+0,00051
+0,00000	+0,00088	+0,01681	+0,02967	+0,01681	+0,00088	+0,00000

+0,00000	-0,00050	-0,01386	-0,02858	-0,01386	-0,00050	+0,00000
+0,00050	+0,00000	-0,02908	-0,03246	-0,02908	+0,00000	+0,00050
+0,01386	+0,02908	+0,00000	+0,00000	+0,00000	+0,02908	+0,01386
+0,02858	+0,03246	+0,00000	+0,00000	+0,00000	+0,03246	+0,02858
+0,01386	+0,02908	+0,00000	+0,00000	+0,00000	+0,02908	+0,01386
+0,00050	+0,00000	-0,02908	-0,03246	-0,02908	+0,00000	+0,00050
+0,00000	-0,00050	-0,01386	-0,02858	-0,01386	-0,00050	+0,00000

+0,00000	+0,00088	+0,01025	+0,00000	-0,01025	-0,00088	+0,00000
+0,00088	+0,04016	+0,03907	+0,00000	-0,03907	-0,04016	-0,00088
+0,01025	+0,03907	+0,00161	+0,00000	-0,00161	-0,03907	-0,01025
+0,00000	+0,00000	+0,00000	+0,00000	+0,00000	+0,00000	+0,00000
-0,01025	-0,03907	-0,00161	+0,00000	+0,00161	+0,03907	+0,01025
-0,00088	-0,04016	-0,03907	+0,00000	+0,03907	+0,04016	+0,00088
+0,00000	-0,00088	-0,01025	+0,00000	+0,01025	+0,00088	+0,00000

6.2.2. ábra. Az a_0, a_1, b_1, a_2 és b_2 paraméterekhez (ebben a sorrendben) tartozó konvolúciós szűrők $r = 2,5\tau$ és bilineáris interpoláció esetén.

Bár a 6.2.5-ben nem így lett jelölve, de a P_i értékek tulajdonképpen függvények, értékük függ a terepponttól (a vizsgált pozíció vízszintes koordinátáitól és a domborzatmodelltől) és a konvolúciós szűrő előállításakor alkalmazott sugártól (mindezek függvényei az a_i és b_i értékek), valamint a kifejezések számításához bevezetett e értéktől.

6.3. Terepszerkezeti formák elkülönítése fuzzy alapokon

A fuzzy logika lényege az, hogy az egyértelmű IGAZ vagy HAMIS érték mellett átmeneti állapotokat is képes kezelni; egy logikai változó így egy 0 és 1 közötti számmal lesz leírható, ahol a 0 az egyértelműen HAMIS, az 1 pedig az egyértelműen IGAZ értéknek felel meg. A klasszikus logikai értékekhez hasonlóan a fuzzy logika értékeivel is lehet logikai műveleteket (ÉS, VAGY) végezni, amire többféle megoldás (az ún. t- illetve s-normák) is létezik. Ezek mindegyikére elmondható, hogy a 0 és az 1 értékekkel számolva a nekik megfelelő klasszikus logikai műveletek eredményét kapjuk, csak az átmeneti értékek esetében működnek különbözőképpen.

A klasszikus („éles”) halmazok esetében definiálhatunk minden halmazhoz egy karakterisztikus függvényt (jelölése általában μ_{halmaz}), ami az alaphalmaz elemeihez logikai IGAZ (1) értéket rendel, ha azok elemei a vizsgált halmaznak, HAMIS (0) értéket pedig, ha nem. Az előbbi mintájára egy halmaz karakterisztikus függvénye egy 0 és 1 közötti értéket (egy fuzzy logikai értéket) is rendelhet az alaphalmaz elemeihez; ilyenkor fuzzy halmazról beszélünk. Ahogyan a fuzzy logikában nem húzunk éles határt az IGAZ és a HAMIS közé, a fuzzy halmazok sem rendelkeznek éles határokkal. [143, 69, 61]

A fuzzy logika és a fuzzy halmazok számtalan helyen használhatóak jól, ahol kizárólag IGAZ vagy HAMIS állításokkal vagy éles kategóriák felállításával túlzottan leegyszerűsítenénk a modellezni kívánt dolgok természetét. A terepszerkezeti formák elkülönítésekor is ez a helyzet, hiszen az ezekhez rendelhető területek sem különülnek el élesen egymástól.

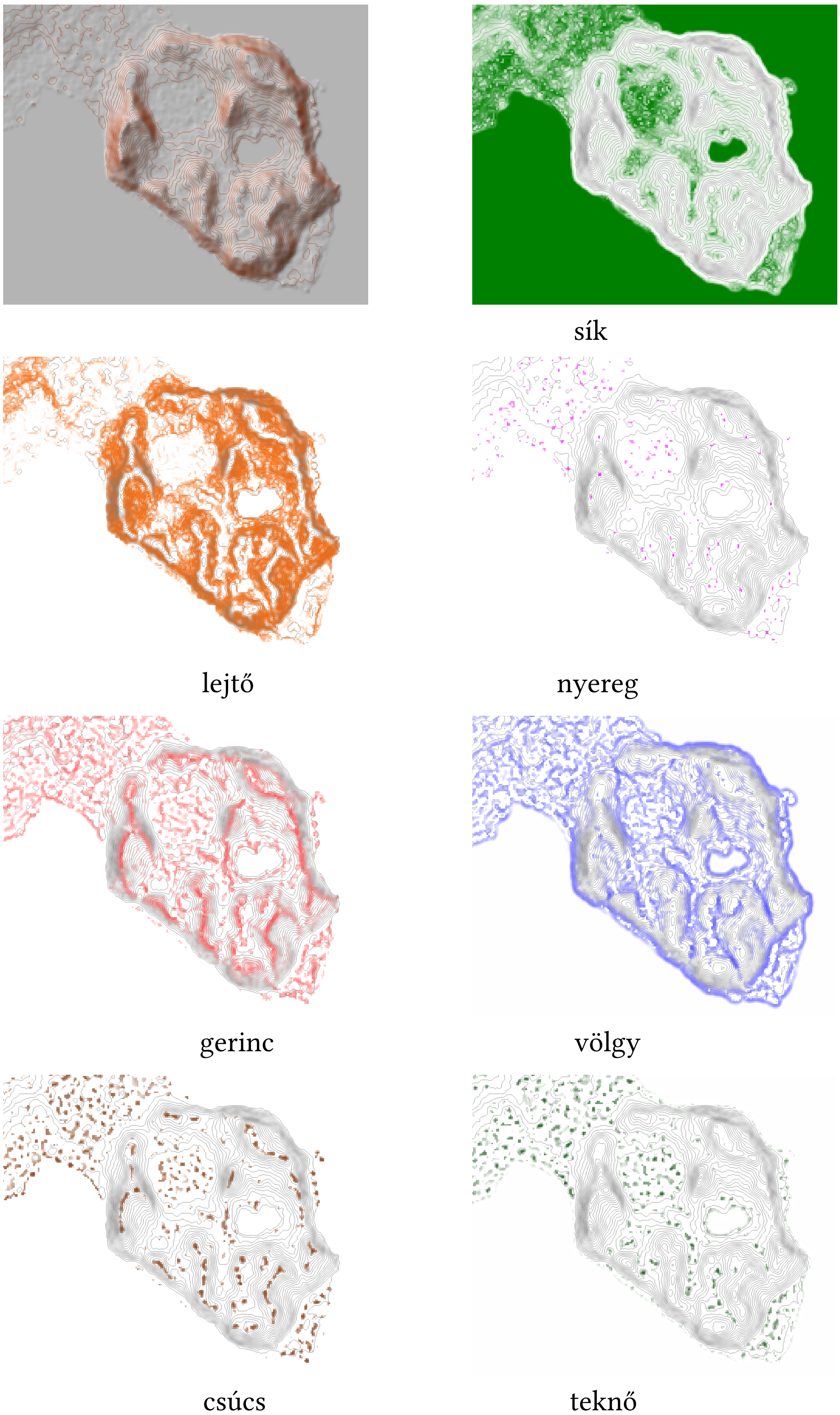
A 6.2.5-ben bevezetett paraméterekből kiindulva az egyes terepszerkezeti formák karakterisztikus függvényeit a következőképpen definiálhatjuk:

$$
\begin{aligned}
\mu_{sík}\,(tereppont) &= P_0^{w_{sík}} \\
\mu_{lejtö}\,(tereppont) &= \begin{cases} 0 & ha\, P_1 < 0 \\ P_1^{w_{lejtö}} & ha\, P_1 \geqq 0 \end{cases} \\
\mu_{gerinc}\,(tereppont) &= \begin{cases} 0 & ha\, P_2 > 0 \\ (-P_2)^{w_{gerinc}} & ha\, P_2 \leqq 0 \end{cases} \\
\mu_{völgy}\,(tereppont) &= \begin{cases} 0 & ha\, P_2 < 0 \\ P_2^{w_{völgy}} & ha\, P_2 \geqq 0 \end{cases} \\
\mu_{csúcs}\,(tereppont) &= \begin{cases} 0 & ha\, P_3 < 0\; vagy\; a_0 \geqq 0 \\ P_3^{w_{csúcs}} & ha\, P_3 \geqq 0\; és\; a_0 < 0 \end{cases} \\
\mu_{teknö}\,(tereppont) &= \begin{cases} 0 & ha\, P_3 < 0\; vagy\; a_0 \leqq 0 \\ P_3^{w_{teknö}} & ha\, P_3 \geqq 0\; és\; a_0 > 0 \end{cases} \\
\mu_{nyereg}\,(tereppont) &= \begin{cases} 0 & ha\, P_4 < 0 \\ P_4^{w_{nyereg}} & ha\, P_4 \geqq 0 \end{cases}
\end{aligned} \tag{6.3.1}
$$

A fenti képletekben a $w_{sík}$, $w_{lejtö}$, w_{gerinc}, $w_{völgy}$, w_{nyereg}, $w_{csúcs}$ és $w_{teknö}$ paraméterekkel az adott terepszerkezeti forma kiemelhető, ha az értékük kisebb egynél. Egynél nagyobb értékekkel az adott terepszerkezeti forma tompítására van lehetőség.

A bemutatott módszereket egy egy másodperces SRTM modellből levezetett, 25 méteres felbontással EOV rendszerben újramintavételezett domborzatmodellen próbáltam ki az $550000 < Y_{EOV} < 650000$ és a $150000 < X_{EOV} < 250000$ koordinátákkal határolt területen, a QGIS és a SAGA programok segítségével. A 6.2.2. ábra konvolúciós szűrőivel képeztem az a_i és b_i paramétereket, a sugár így $R = 62.5\,m$ volt (a $\tau = 25\,m$ felbontás két és félszerese). Az érdességi tényezőnek $e = 1m$-t vettem fel, a kiemelési tényezők $w_{sík} = 1, w_{lejtö} = 1, w_{gerinc} = 0.5, w_{völgy} = 0.5, w_{nyereg} = 0.25, w_{csúcs} = 0.5$ és $w_{teknö} = 1$ voltak. Az elemzések eredményét a Tihanyi-félszigetre vonatkozóan a 6.3.1. ábra mutatja.

A terepszerkezeti formák fuzzy alapú elkülönítése sokféle térbeli fuzzy elemzésnek lehet az alapja. Az adatok defuzzyfikálásával akár a terepszerkezeti formák klasszikus osztályozását is elő lehet állítani.



6.3.1. ábra. A különféle terepszerkezeti formákhoz rendelhető fuzzy értékek.

7. fejezet

Domborzatmodellek létrehozása pontfelhők alapján

A légi lézerszkenneléssel (LiDAR) nyert adatok feldolgozása esetén az egyik legfontosabb végtermék a felmért terület digitális domborzatmodellje. A pontfelhő alapján úgy kell meghatároznunk a terep felszínét, hogy a terep felett elhelyezkedő tárgyakon keletkezett pontok a végeredményt ne befolyásolják. Olyan robusztus módszerre van ilyenkor szükségünk, amelyet az sem zavar, ha a pontok túlnyomó része a terepfelszín felett helyezkedik el.

Ezt általában olyan megoldásokkal érik el, amelyek egy szűrési lépésben eltávolítják a nem a terepfelszínhez tartozónak ítélt pontokat, majd a szűrt pontfelhőre illesztik a domborzatmodellt, vagy a létrehozott domborzatmodell felületén utólag hajtanak végre valamilyen szűrési műveltet. Ebben a fejezetben ismertetek egy olyan módszert, amelyik a pontfelhő előzetes szűrése nélkül képes a terepfelszín magasságát egy tetszőleges pozícióban meghatározni, amit azután többféle módon lehet felhasználni domborzatmodellek előállítására vagy a terep egyes objektumainak felismerésére. (4. tézis)

7.1. Felhasználható elvek, lehetséges megoldások

A probléma megoldása során támaszkodhatunk arra, hogy bár a pontok nagy hányada képződhet a terepfelszín felett (fákon, bokrokon vagy egyéb objektumokon), a terepfelszín alá legfeljebb mérési hibából eredően kerülhetnek (és a tapasztalatok alapján kerülnek is) pontok. A terepfelszín kiértékelésekor ezért a legalacsonyabban elhelyezkedő összefüggő felszínt kell keresni. Többféle lehetőség kínálkozik ilyen felületek keresésére, amelyeket az alábbiakban fogok bemutatni.

7.1.1. Feldolgozási módszerek a gyakorlatban

A légi lézerszkenneres mérésekből kapott adatok feldolgozásakor az egyik lehetőség az, hogy a pontfelhőben található nagy számú, nem a terepfelszínhez tartozó pontot valamilyen szűréssel megpróbálhatjuk kizárni a későbbi műveletekből. A [137] által javasolt lejtés alapú szűrő (slope-based filter) kiválasztja azokat a pontokat, amelyeknél egy a csúcsával a pontra illesztett függőleges tengelyű kúpon belül nem található a pont alatt a pontfelhőnek további pontja. A kúp fél nyílásszögének pótszöge határozza meg azt a maximális lejtést, amit a szűrt pontfelhő által leírt felületen belül még elfogadhatónak tartunk, az ennél meredekebb részeket eredményező pontokat (ezek tipikusan a fákon vagy az épületeken képződnek) a szűrés eltávolítja; a szűrés eredményeként kapott pontfelhőben nem lehetnek olyan pontpárok, amelyeket ennél meredekebb szakasszal lehetne összekötni. A módszert továbbgondolva [122] egy olyan megoldást javasol, ahol a szűréshez használt lejtés a terephez alkalmazkodva dinamikusan változik.

Egy másik lehetőség a LiDAR adatok feldolgozására, hogy a pontfelhőre illesztett felületmodellen végzünk valamilyen szűrési műveletet a nem a terepfelszínhez tartozó pontok hatásának kiküszöbölésére. Ilyenek az [55]-ben javasolt morfológiai szűrő vagy a [89]-ben bemutatott módszerek.

A morfológiai szűrők koptatási (erosion) és bővítési (dilatation) lépésekből épülnek fel, amelyeknek lényege az, hogy egy a kép egy pontjához a környező területek legalacsonyabb vagy legmagasabb értékét rendelik hozzá. A környező terület méretét és alakját egy ablak (window) jellemző határozza meg. A kétfajta művelet kombinálásával áll elő egy úgynevezett dual rank szűrő (Dual Rank Filter).

A [144]-ban bemutatott és a gyakorlatban széles körben használt progresszív morfológiai szűrő (Progressive Morphological Filter) alapelve az, hogy a morfológiai szűréssel kapott felületmodellt használjuk egy következő lépésben a pontfelhő szűrésére azon az elven, hogy a terepfelszínhez tartozó pontok a szűréssel kapott felület közelében helyezkednek el. Az ezt követő lépésben az így szűrt pontfelhőre illesztünk egy újabb felületet, amit aztán ismét egy morfológiai szűrőn engedünk át, hogy utána a pontfelhő egy következő szűréséhez használhassuk. Ezeket a lépéseket úgy ismételjük egymás után, hogy a morfológiai szűrő által használt ablak méretét egyre kisebbre vesszük.

Kifejezetten az erdős területekről LiDAR adatok alapján készített domborzatmodellek problémájával foglalkozik [106] és [85] mellett [70] is, ahol a feladatra egy virtuális erdőirtásnak (Virtual Deforestation, rövidítve VDF) nevezett módszert mutatnak be.

A [145] egy ötletes módszert mutat be a domborzatmodellek LiDAR adatokból történő előállítására, ami a számítógépes animációkhoz használt ruha szimuláción alapul. A ruha szimulációt a számítógépes animációkban arra használják, hogy a különféle szöveteknek, mint például a szereplők ruhái, egy zászló vagy egy függöny, a mozgását modellezzék. Ez a modellezés szorosan kapcsolódik más fizikai modellezésekhez, így a megadott fizikai tulajdonságokkal (tömeg, merevség) rendelkező szövet mechanikai kölcsönhatásba léphet a színtér egyéb elemeivel; például a ruha ráfeszül a szereplőre, aki viseli; a függöny pedig, miközben fújja a szél, nekiverődik a falnak.

Domborzatmodell előállítására a fenti eszközt úgy lehet felhasználni, hogy az animációs program színterében egy szövetet rádobnak a lefelé fordított pontfelhőre. Ezzel egyenértékű lehet az a megoldás is, ha a pontfelhőt nem fordítjuk meg, de a nehézségi erő vektorát (ha az adott programban az szabályozható) a szokásossal ellentétesen felfelé, a Z tengely irányába mutatónak állítjuk be. Ha a pontfelhő pontjai mechanikai akadályt képeznek, akkor a rajtuk felfekvő szövet alakja jól fogja visszaadni a domborzat felületét.

A különféle szűrési módszerek összehasonlításával [123] foglalkozik. Többféle módszer részletes bemutatása olvasható a [98]-ban. A [80] a LiDAR adatok többprocesszoros környezetben való feldolgozásával foglalkozik.

Létezhetnek a felszín kiértékelésének speciális esetei is. Ilyen például, amikor egy útburkolat felszínét szeretnénk nagy pontossággal meghatározni [142].

7.1.2. Legalacsonyabb rész kiválasztása

A pontfelhő egy adott területre (pl. az a rész, aminek egy pozíciótól mért vízszintes távolsága kisebb egy meghatározott sugárnál) eső darabjának meghatározzuk a legalacsonyabb részét, és ezt tekintjük a terepfelszín magasságának az adott terület középpontjában. A legalacsonyabb rész meghatározásakor a legkisebb magasságú pont helyett érdemes annak a pontnak a magasságát venni, ami a többi a pontfelhő pontjainak egy meghatározott részénél (például 1-10 százalékánál) magasabban van, hogy az esetleg durva hibával terhelt pontokat így kiszűrjük.

A műveletet több pozícióban elvégezve, például az említett függőleges henger tengelyét egy rácsháló pontjaihoz illesztve digitális domborzatmodellt tudunk készíteni. A módszer hátránya, hogy a vizsgált terület magasságát a legalacsonyabb részek alapján fogja meghatározni, ami miatt ez a megoldás a lejtős területeknél nyilvánvalóan nem használható.

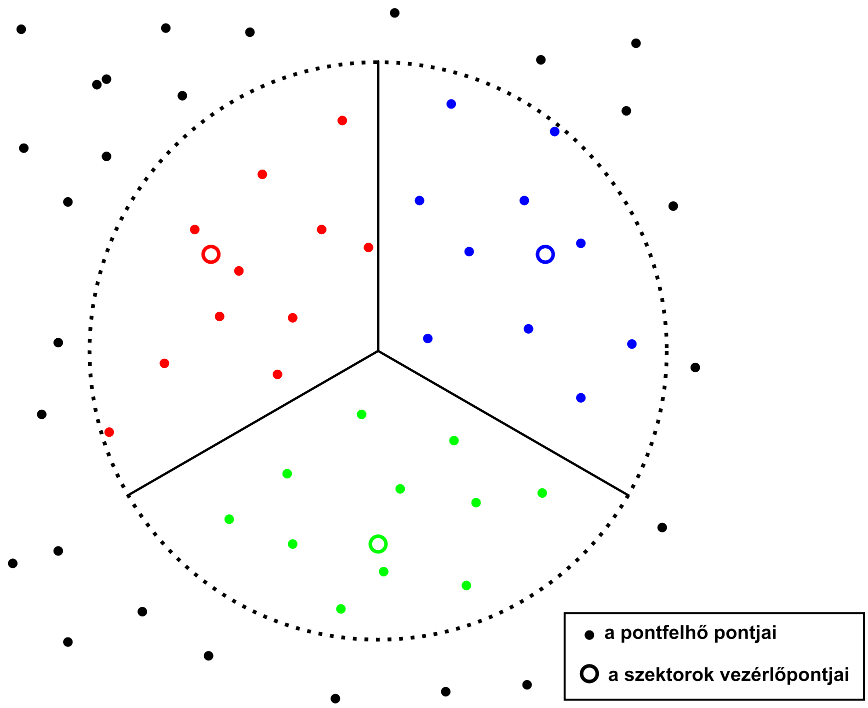
7.1.3. Sík illesztése

Az előbb bemutatott módszer hátránya abból ered, hogy a vizsgált területeken a terepfelszínt vízszintes síknak tekintjük, holott sok esetben nyilván nem az. Ennek a következménye az, hogy lejtős terep esetén a kapott érték nem a terület átlagos magasságát vagy középpontjának magasságát fogja jelenteni, hanem a legkisebb, tipikusan valahol a terület szélén elhelyezkedő magasságot. (Feltételezve hogy egy olyan vízszintes síkot helyezünk el, amely alatt a pontoknak csak egy kis hányada található.) Ezt a 7.1.2 ábra bal alsó képén is ábrázolt helyzetet úgy küszöbölhetjük ki, ha vízszintes sík helyett megpróbálunk egy általános helyzetű síkot alkalmazni.

A legalacsonyabb rész egyszerű kiválasztásakor az egy paraméterrel (a magasságával) megadható vízszintes sík meghatározásához egyetlen feltételt adtunk meg: a sík alatt helyezkedjen el a pontok meghatározott hányada. Egy általános helyzetű sík három paraméterrel adható meg, így a meghatározásához is három feltételre van szükség. Az egyetlen feltételünkből úgy tudunk hármat csinálni, ha a területet három egyenlő részre osztjuk, és mindhárom részterület esetében előírjuk, hogy a pontok meghatározott részének a sík alatt kell elhelyezkednie [8].

Ha a vizsgált pozíció körül egy kör alapterületű területen gyűjtöttük ki a pontfelhő pontjait, ezt a területet három egymással 120 fokot bezáró sugár mentén három egyforma körcikkre tudjuk osztani a 7.1.1 ábrának megfelelően. Az illesztendő sík három paraméterét az ezen három körcikk vezérlőpontjaiban vett magasságokkal adjuk meg. A vezérlőpontok a szektorok felezővonalában, a szektor csúcsától kétharmad sugárnyi távolságban helyezkednek el.

Kezdeti lépésként mindhárom vezérlőpont magasságát úgy vesszük fel, hogy a hozzájuk tartozó körcikkekre eső pontok q-ad része ($q \cdot 100$ százaléka) legyen annál alacsonyabban. A továbbiakban sorra megyünk az egyes vezérlőpontokon, és a magasságukat úgy növeljük vagy csökkentjük, hogy a hozzájuk tartozó (a körcikkük területére eső) pontok q-ad része kerüljön a három vezérlőpont által meghatározott sík alá. A műveletet addig



7.1.1. ábra. A vizsgált pozíció R sugarú környezetét az onnan gyűjtött pontokkal együtt három szektorra osztjuk. A kigyűjtött pontokat az ábrán szektoronként különböző színnel jelöltem. A műveletek során szerepet kapnak még a szektorok vezérlőpontjai, amelyeket a szektor színének megfelelő körök jelölnek.

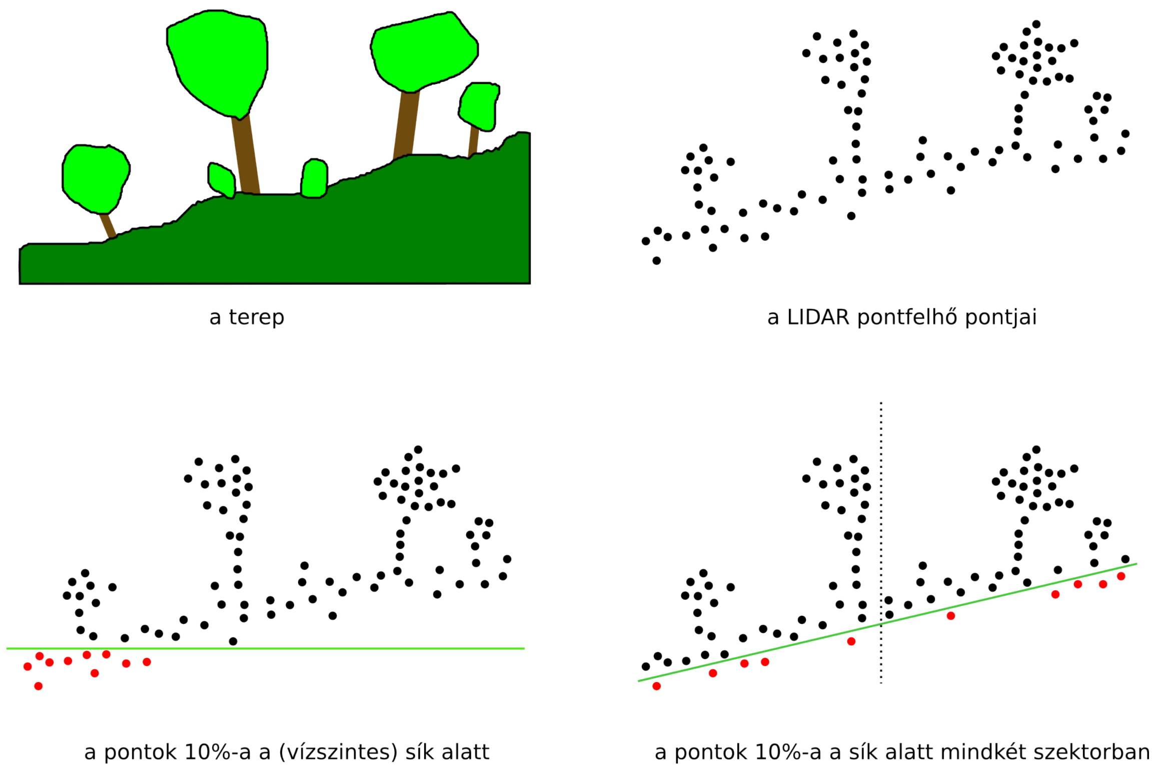
ismételjük, ameddig a súlypontok magasságának változtatása nélkül mindhárom körcikk pontjainak egy hibahatáron belül q-ad része lesz a sík alatt.

A bemutatott módszer nem csupán egy magasságot határoz meg egy vízszintes pozícióban, hanem az illeszkedő sík dőlésére vonatkozó adatokat is megadja. A három adat, amivel a síkot meghatározzuk, lehet akár a síknak vizsgált pozícióban vett magassága és az alkalmazott koordináta-rendszer vízszintes tengelyeinek irányába eső lejtései.

A bemutatott elvet a három dimenziós tér helyett egy két dimenziós síkban is használhatjuk (ld. még 7.7.2). A módszer lényege így könnyebben megérthető és szemléltethető. Egy vonalat úgy tudunk egy ponthalmazra illeszteni, hogy a területet (vagy annak egy vizsgált részterületét) függőlegesen két részre osztva mindkét féltéren a pontok q-ad része kerüljön a vonal alá. (7.1.2 ábra)

7.1.4. Sík illesztésének korlátai

A bemutatott módszer abból a feltételezésből indul ki, hogy a kiértékelendő felület a vizsgált pozíció közelében jó közelítéssel egy dőlt sík. Ez általában sokkal jobb közelítés annál, mint amikor a terephez egy vízszintes síkot próbálunk illeszteni, de ha a vizsgált pozíció megadott sugarú környezetében a felületnek jelentős görbülete van, akkor az illesztett sík nem tudja azt megfelelően reprezentálni. Ha csökkentjük a vizsgált felületdarab méretét (vagyis csökkentjük a vizsgált pozíció körüli kör sugarát), egy síkkal jobban közelíthető ponthalmazt fogunk kapni, de ezzel együtt csökkenni fog az illesztéshez használható pontok száma is.



7.1.2. ábra. A módszer elvének szemléltetése két dimenzióban. A bal felső kép a terep valós megjelenését próbálja bemutatni, a mellette található kép a egy LiDAR pontfelhőnek az ezen a terepen képződött pontjait mutatja. A bal alsó részen egy vízszintes síkot (itt most a dimenziószám csökkentése miatt egy egyenest) illesztünk a pontfelhőhöz úgy, hogy a pontok 10 százaléka kerüljön a vonal alá. A jobb alsó kép a javasolt illesztési módszer kétdimenziós megfelelőjét mutatja be, ami úgy helyez el egy ferde egyenest, hogy a két szektorra osztott pontfelhő mindegyik részében a pontok 10 százaléka kerüljön a vonal alá, amivel így a terepfelszínnek egy jó közelítését kapjuk annak ellenére, hogy a pontok jelentős része a fákon és a bokrokon keletkezett.

7.2. Gyakorlati megvalósítás

A bemutatott algoritmust angolul „Fitting Disc Method”-nak neveztem el (a megnevezésben a korong arra utal, hogy a sík illesztését lokálisan, egy kör alakú területen végezzük.), és a gyakorlatban Python[1] nyelven írt alkalmazásokkal valósítottam meg. Az alkalmazások alapját képező `fitdisc` modul tartalmaz egy `Ptcloud` osztályt, aminek a `readptfile` metódusa képes egyszerű szöveges állományokból a pontfelhő pontjait beolvasni a pontfelhő objektumba, és egy `fitdisc` metódust is, ami egy vízszintes koordinátáival megadott pozícióban egy síkot illeszt a pontfelhő objektum pontjaira. Mivel előzetesen a vizsgált pozíció adott sugarú környezetéből gyűjtünk pontokat, az illesztett sík tulajdonképpen egy korongnak tekinthető.

A sík illesztését végző metódus belül egy helyi koordináta-rendszert használ a számításokhoz. Ehhez a pontokat eltoljuk vízszintesen úgy, hogy a koordináta-rendszernek a kezdőpontja a vizsgált pozícióba essen, majd olyan mértékű nagyítást alkalmazunk, hogy a pontok kigyűjtéséhez használt kör sugara pontosan egységnyi legyen. A pontfelhő koordináta-rendszeréből (nagy betűkkel jelölve) a következő képletekkel térhetünk át ebbe a (kis betűkkel jelölt) helyi rendszerbe:

$$\begin{aligned} x &= \tfrac{X-X_0}{R} \\ y &= \tfrac{Y-Y_0}{R} \\ z &= Z \end{aligned} \tag{7.2.1}$$

Az X_0 és az Y_0 a vizsgált pozíció koordinátái, az R pedig a pontok gyűjtéséhez használt kör sugara. (A pontfelhő egy pontja akkor vesz részt az illesztésben, ha vízszintes értelemben a pozíció R sugarú környezetében helyezkedik el, vagyis $R^2 < (X - X_0)^2 + (Y - Y_0)^2$) A magasságokon a program nem változtat, a sík illesztéséhez használt helyi koordináta-rendszerben is az eredeti magasságokkal dolgozunk, így majd az eredményt is ebben a rendszerben kapjuk.

A síkot kétféle módon is meg lehet határozni. Az egyik lehetőség a síknak az előbb bemutatott helyi koordináta-rendszerben felírt egyenlete:

$$z = z_0 + ax + by \tag{7.2.2}$$

Ebben az egyenletben a z_0 érték a sík magassága a helyi koordináta-rendszer kezdőpontjában, vagyis a vizsgált pozícióban. Az a és b paraméterek az illesztett síknak a dőlését fejezik ki, de ha az eredeti koordináta-rendszerben értelmezhető értékeket akarunk belőle kapni, akkor még osztanunk kell őket R-el. Az 7.2.2 segítségével a z_0, a és b paraméterek ismeretében gyorsan ki tudjuk számítani a sík egy pontjának magasságát egy vízszintes koordinátáival adott pozícióban, majd ezt követően el tudjuk dönteni, hogy a pontfelhő egy azonos vízszintes helyzetű pontja ehhez képest hogyan helyezkedik el (alatta vagy felette van-e a síknak).

A másik lehetőség a sík meghatározására, hogy a síknak a három körcikk vezérlőpontjában vett magasságait adjuk meg, amelyekre a továbbiakban z_{w_0}, z_{w_1} és z_{w_2} jelö-

[1] A Python programozási nyelv honlapja a https://www.python.org/ címen érhető el.

lésekkel hivatkozunk. (A z_{w_i} jelölésekben az i a szektorokat azonosító nulla és kettő közötti sorszám.) Ez ugyanúgy három adatot jelent, mint a sík klasszikus egyenletének paraméterei, de a későbbiekben előnyösebb lesz számunkra az illeszkedő sík adatainak keresésekor, mert az egyik ilyen magasságnak a megváltoztatása jobban befolyásolja a hozzá tartozó szektorban elhelyezkedő pontoknak a síkhoz viszonyított helyzetét, mint a másik két szektor pontjaiét.

A vezérlőpontok magasságaiból számíthatóak az egyenlet paraméterei:

$$\begin{aligned} z_0 &= \tfrac{z_{w_0}+z_{w_1}+z_{w_2}}{3} \\ a &= \tfrac{3\sqrt{3}(z_{w_2}-z_{w_0})}{4} \\ b &= \tfrac{3(z_{w_1}-z_0)}{2} \end{aligned} \tag{7.2.3}$$

Visszafele a számítás még egyszerűbb, mert csak a körcikkek vezérlőpontjainak a koordinátáit kell az 7.2.2 egyenleteibe behelyettesíteni:

$$\begin{aligned} z_{w_0} &= z_0 - \tfrac{2\sqrt{3}a}{9} - \tfrac{2b}{9} \\ z_{w_1} &= z_0 + \tfrac{4b}{9} \\ z_{w_2} &= z_0 + \tfrac{2\sqrt{3}a}{9} - \tfrac{2b}{9} \end{aligned} \tag{7.2.4}$$

Kezdő lépésként a program minden szektor esetében a szektor területére eső ponthalmazra megállapítja azokat a magasságokat, amelyek alatt a pontok q-ad része helyezkedik el. Ezek a magasságok lesznek a z_{w_0}, z_{w_1} és z_{w_2} értékek kezdőértékei. (Értelemszerűen az adott szektor vezérlőpontjához tartozó értékek.)

A következő lépésben úgy kell megváltoztatni a soron következő z_{w_i} értékét, hogy a hozzá tartozó körcikkben a pontok q-ad része kerüljön a z_{w_0}, z_{w_1} és z_{w_2} értékek által meghatározott sík alá. (A másik két $z_{w_{(i\pm1)\ mod\ 3}}$ érték értelemszerűen változatlan marad.) A változtatásokat úgy végezzük, hogy a z_{w_i} értéke egy t érték (alapértelmezett értéke a programban 0.01) egész számú többszöröse legyen. Ezt a műveletet kell folytatni a soron következő $z_{w_{(i+1)\ mod\ 3}}$értéken, egészen addig, ameddig három egymást követő esetben nem kell megváltoztatni a z_{w_i} értéket. A végleges z_{w_0}, z_{w_1} és z_{w_2} értékekből az 7.2.3 felhasználásával számított z_0 lesz a pont magassága. Ha szükségünk van rá, akkor a sík dőlésére vonatkozó adatokat is számíthatjuk az $\frac{a}{R}$ és a $\frac{b}{R}$ összefüggésekkel.

A sík és a pontok helyzetének vizsgálatakor a program három kategóriát különböztet meg, és megszámlálja az ezekbe tartozó pontokat. A sík alatti (n_{-1}) és a sík feletti pontok (n_{+1}) mellett külön számoljuk a síkhoz közeli pontokat (n_0), melyek magasságának az eltérése a sík velük azonos vízszintes helyen vett magasságával $1.6 \cdot t$-nél kevesebb. Az sík alatti pontokra előírt q arányt akkor tekintjük teljesítettnek, ha a sík alatti pontok aránya kisebb, a sík alatti és a sík közeli pontok összesített aránya pedig nagyobb nála:

$$\frac{n_{-1}}{n_{-1}+n_0+n_{+1}} \leq q \leq \frac{n_{-1}+n_0}{n_{-1}+n_0+n_{+1}} = 1 - \frac{n_{+1}}{n_{-1}+n_0+n_{+1}} \tag{7.2.5}$$

Ha a q ezen az intervallumon kívül van, akkor szükség van a körcikkhez tartozó tartozó pont z_{w_i}magasságának növelésére (ha a sík alatti és közeli pontok aránya kisebb p-nél) vagy csökkentésére (ha a sík alatti pontok aránya nagyobb q-nál). Ezt úgy végezzük el, hogy t egységnyi változtatással kezdve addig duplázzuk meg a változtatás mértékét, ameddig az eltérés már nem lesz azonos irányú a kezdetivel. (Ellentétes irányú lesz, vagy éppen egy megfelelő értéket találunk.) Az utolsó és az utolsó előtti változtatás között ezt követően felező módszerrel tudunk egy megfelelő értéket keresni.

A programmal lehetőség van arra, hogy egy rácsháló minden rácspontjának pozíciójában elvégezve a vizsgálatot felületmodellt állítsunk elő, és azt egy ArcInfo ASCII GRID fájlba kiírjuk. Nem csupán a magasságokat lehet ilyen módon egy állományba elmenteni, hanem az illesztett síkok dőlésére vonatkozó paramétereket is.

7.3. A paraméterekkel kapcsolatos kérdések vizsgálata

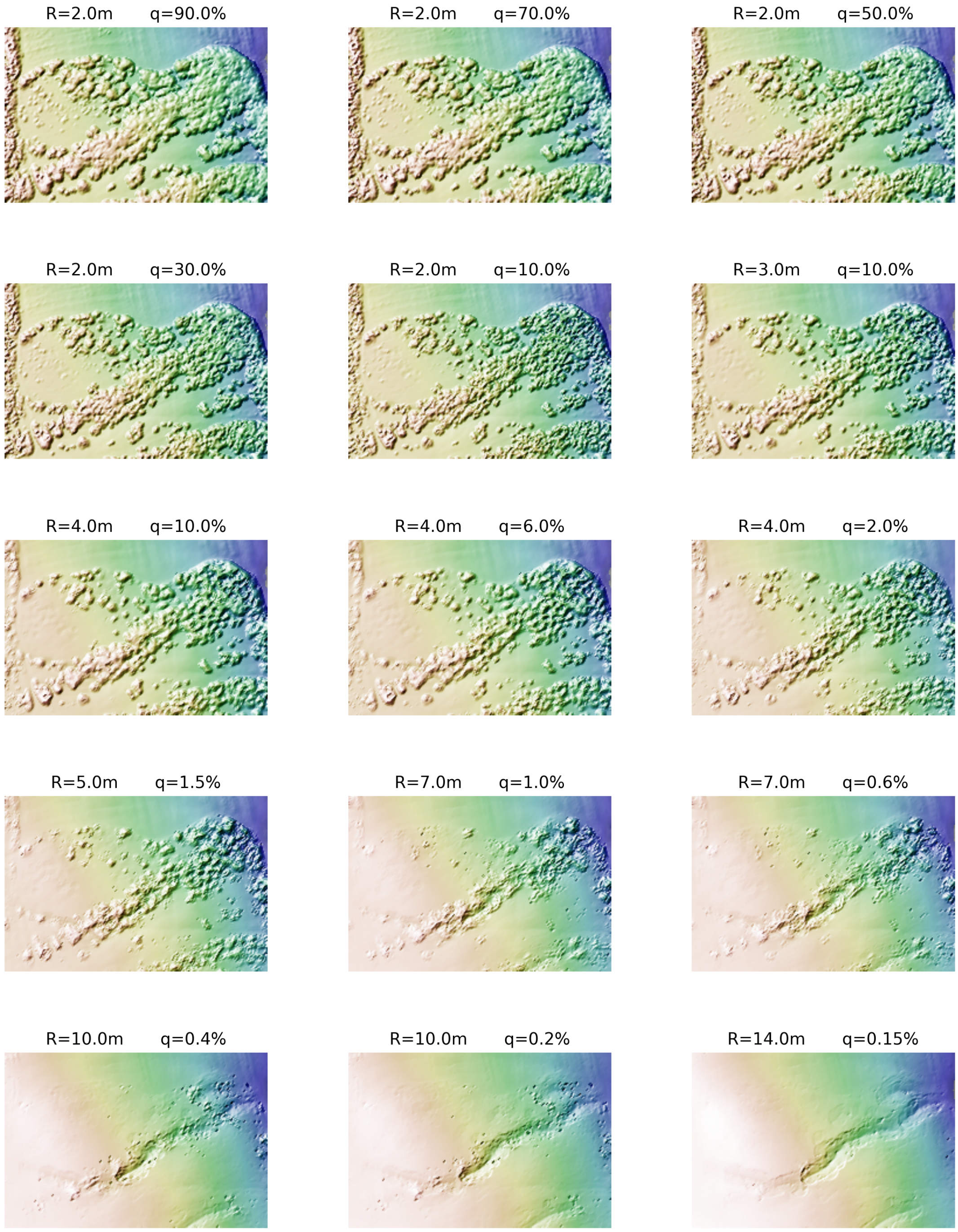
A sík illesztése a bemutatott módon két szabadon felvehető (vagy úgy is mondhatjuk, hogy önkényesen meghatározható) paraméter megadását igényli. Ezek egyike a pontok gyűjtéséhez használt kör R-el jelölt sugara, a másik pedig a q-val jelölt paraméter, ami az illesztett sík alá (vagy közvetlen közelébe) eső pontok arányát adja meg. Különféle R és q értékekkel készült domborzatmodelleket láthatunk a 7.3.1 ábrán.

Az R értékének növelésével egyre több pontunk lesz, így az illesztett sík is pontosabb és megbízhatóbb lesz, viszont a kapott értékek egy egyre nagyobb területet fognak jellemezni, a síkok z_0 magasságaiból képzett felületmodellekről eltűnhetnek az apróbb részletek. Az R értékét felvehetjük dinamikusan, a legkisebb olyan sugárként, ami minden szektor területén tartalmaz legalább N darab pontot. Természetesen a programnak ilyenkor az N értéket kell megadni R helyett, tehát a művelet paramétereinek a száma nem lesz kevesebb, csak az egyik paraméter jelentése lesz más.

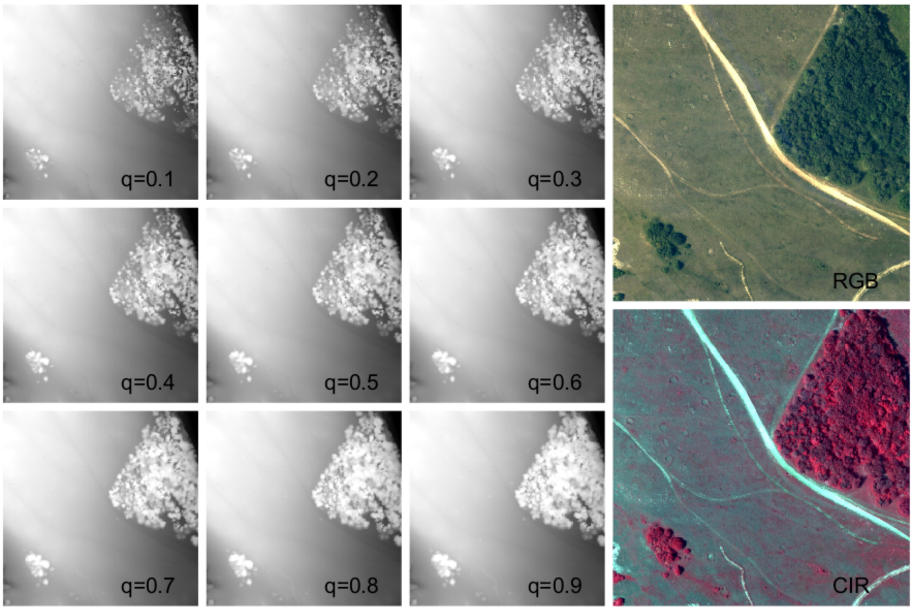
Különböző q paraméterek használatával különböző felületeket fogunk kapni. Minél nagyobb a q értéke a sík annál magasabban helyezkedik el, hiszen annál több pontnak kell alá kerülnie. A különféle paraméterekkel kapott felületek közötti távolság is egy fontos, a pontfelhő elemzésével nyerhető leíró jellemzővé válik. Ha a pontok az illeszthető sík feletti magasság tekintetében elszórtan helyezkednek el, a felületek közötti távolság nagyobb lesz, míg ellentétes esetben kisebb.

Hasonló módon lehet vizsgálni a különféle q értékekkel képzett síkok merőleges irányai közötti eltéréseket. Ha a különféle paraméterekkel generált felületek érintői hasonló irányokba mutatnak, az a pontok egyenletes eloszlására utal. Szintén meg lehet vizsgálni azt, hogy ezek az irányok milyen viszonyban vannak egy levezetett felületmodell alapján a 2.4.2 összefüggéssel számított hasonló értelmű jellemzőkkel.

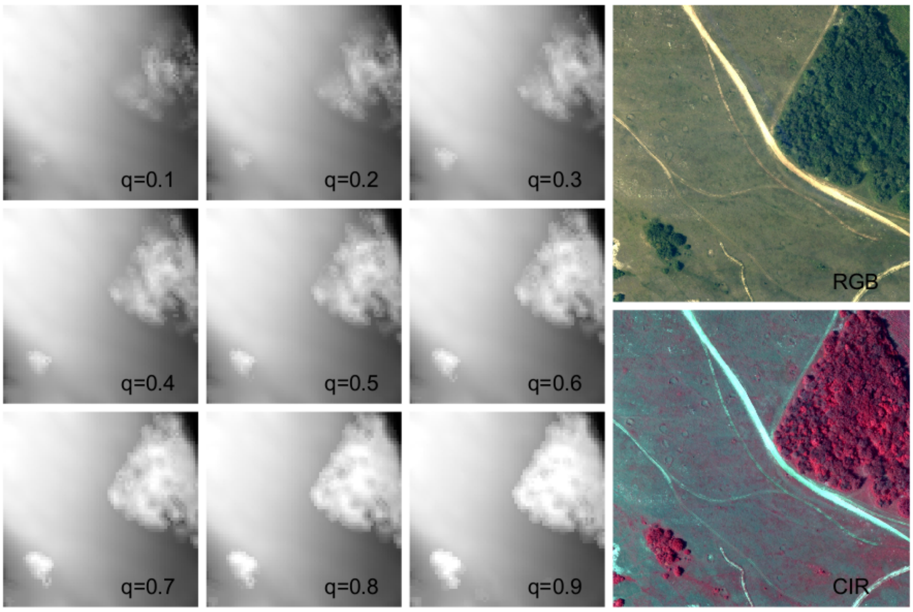
A bemutatott módszer hátrányaként is felfogható, hogy előzetesen meg kell adni egy önkényesen felvehető paramétert (q) arra vonatkozólag, hogy a pontok milyen arányú része kerüljön a sík alá. A pontok eloszlásának tanulmányozásával rugalmasan is megpróbálkozhatunk megállapítani egy ilyen értéket. Létrehozhatunk M darab felületet (pozíciónként számíthatunk M darab magasságot), úgy hogy $q_i = \frac{i}{M+1}$, tehát a q az $\frac{1}{M+1}, \frac{2}{M+1}, \frac{3}{M+1}, \ldots\ldots, \frac{M-1}{M+1}, \frac{M}{M+1}$ értékeket vegye fel. Az így kapott magasságok közül kiválaszthatjuk az egymáshoz legközelebb lévő kettőt, és azok átlagát vehetjük a felület magasságának.



7.3.1. ábra. Egy terület domborzatmodellje különféle R és q értékekkel.



7.3.2. ábra. Különféle q paraméterekkel, $R = 2$ méteres sugárral számított felületek. Az ábra bal szélén az Iszkaszentgyörgy külterületén található terület színhelyes (felül) illetve hamis színes (alul) ortofotója látható.



7.3.3. ábra. Különféle q paraméterekkel, $R = 10$ méteres sugárral számított felületek. Az ábra bal szélén az Iszkaszentgyörgy külterületén található terület színhelyes (felül) illetve hamis színes (alul) ortofotója látható.

7.3.4. ábra. Különféle q paraméterekkel, $R = 2$ méteres sugárral számított felületek. Az ábra bal szélén a Székesfehérvár belterületén található terület színhelyes (felül) illetve hamis színes (alul) ortofotója látható.

7.4. Alkalmazási lehetőségek

A bemutatott módszert elsődlegesen digitális domborzatmodellek előállítására lehet használni LiDAR pontfelhők feldolgozásával. Előnye, hogy a terepfelszín felett a növényzeten és a különféle tereptárgyakon keletkező pontok hatását megfelelő paraméterezéssel hatékonyan kiszűri, feltéve hogy elégséges számú pont képződik a terepfelszínen.

A különféle paraméterekkel illesztett síkok magasságai közötti különbségek, illetve az illesztett síkok merőleges irányai közötti eltérések mértéke olyan a pontfelhő pontjainak a vizsgált hely környezetében vett eloszlását kifejező számok, amelyek alkalmasak lehetnek a terepfelszín jellegének a vizsgálatára, ha a multispektrális felvételek adataihoz hasonlóan kezeljük őket.

7.4.1. Domborzatmodellek létrehozása

A bemutatott módszer bármilyen vízszintes koordinátákkal megadott pozícióhoz képes a pontfelhő alapján egy magasságot rendelni. Ezek a pozíciók lehetnek egy szabályos négyzetrácsháló rácspontjai is, így egyszerűen elő lehet állítani egy GRID domborzatmodellt. A módszerrel kapható egyéb jellemzők (például a sík lejtésére vonatkozó adatok, vagy a különféle q és R értékekkel kapott eredmények eltérései) alapján is hasonló módon lehet raszter állományokat előállítani.

TIN domborzatmodellek előállítására is használható lehet a bemutatott módszer a [70]-ben leírt módszerhez hasonlóan.

7.4.2. Erdős és bokros területek vizsgálata

Az erdős területek esetében a pontoknak csak nagyon kis hányada képződik a terepfelszínen. Ez komoly kihívást jelent a feldolgozóprogramok számára a domborzatmodellek előállításakor.

Megfelelő körülmények között (kellően alacsony q érték, illetve olyan kellően nagy R érték, hogy megfelelő számú pont kerüljön az egyes szektorokba) a bemutatott módszer is alkalmas arra, hogy erdős vagy bokros területeken a terepfelszín magasságát megállapítsuk a segítségével egy tetszőleges pozícióban, így erdős területek domborzatmodelljeit is elő tudjuk vele állítani. Fontos megjegyezni, hogy csodára ez az eljárás sem képes, ha növényzet túl sűrű és ennek következtében nem jut elég pont a terepfelszínre, akkor nyilván nem tudjuk megállapitani a terepfelszín magasságát. (Pontosabban az eljárás egy attól eltérő értéket határoz meg.)

Az erdős területek domborzatának kiértékelése mellett fontos lehet a munkaterület ilyen jellegű részeinek a pontfelhő pontjainak eloszlása alapján történő felismerése is. Erre alkalmas lehet egy pozícióban különböző q értékekkel kapott magasságoknak és a hozzájuk tartozó síkok dőléseinek a vizsgálata.

7.4.3. Tetők kiértékelése

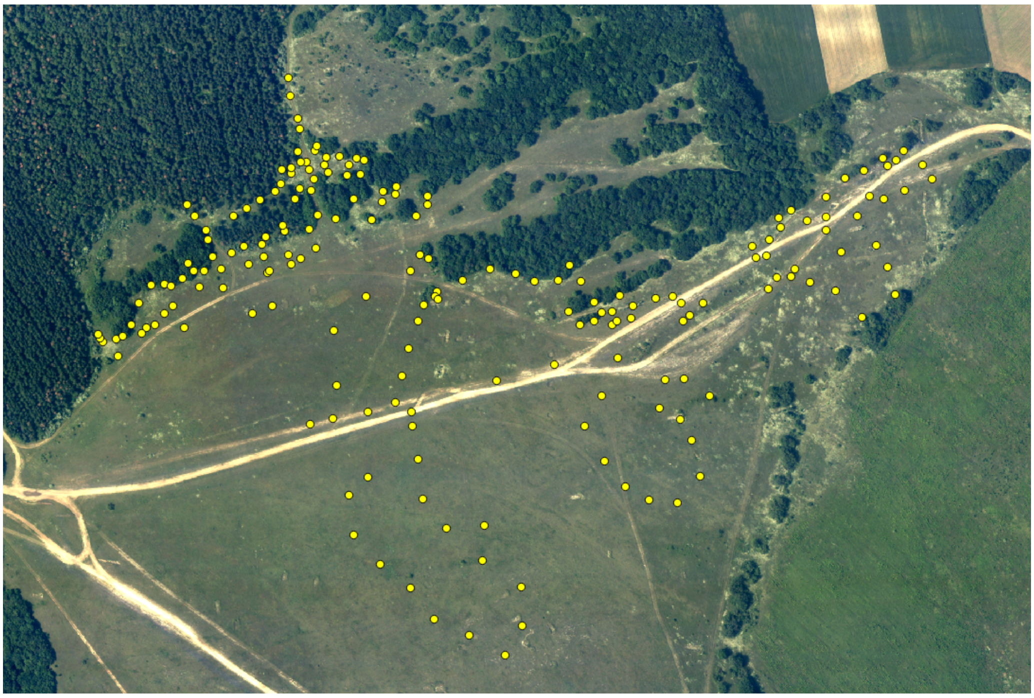
A beépített területekről készült LiDAR méréseken a pontok jelentős része az épületek tetőszerkezetén keletkezik. Ezek általában összefüggő és megközelítőleg sík felületdarabokból állnak. A tetők a fizikai földfelszínhez hasonlítanak abból a szempontból, hogy alattuk már nem keletkeznek pontok, hiszen a mérésekhez használt lézerfény nem tud áthatolni rajtuk.

A tetők kiértékelésével kapcsolatos feladatokra különösen alkalmassá teszi a bemutatott módszert az, hogy a vizsgált pozícióban nem csupán egy magasságot (z_0) ad meg, hanem egy síkot illeszt a vizsgált pozíció környezetében a pontfelhő pontjaira, ami a magasságon túl a sík irányának meghatározását is jelenti.

A létrehozott felületmodellen egymás szomszédságában elhelyezkedő, irányukat is figyelembe véve egy síkra illeszkedő rácspontokat egy tetőidommá lehet összevonni. Az elemzések során kihasználhatjuk, hogy a vizsgálatokat bármilyen vízszintes pozícióban el lehet végezni, nem csak egy rácsháló rácspontjaiban, így a megfelelő részeken sűrűbben is illeszteni lehet síkokat a pontfelhőhöz.

7.5. A módszer gyakorlati alkalmazhatóságának vizsgálata

A bemutatott módszer tanulmányozásához egy mintaterületről rendelkezésre álló LiDAR pontfelhőt többféle módszerrel feldolgoztam. Az érintett területen geodéziai méréseket is végeztünk.



7.5.1. ábra. A módszer ellenőrzése érdekében végzett geodéziai mérés pontjainak elhelyezkedése a terepen.

7.5.1. A vizsgálatokhoz használt LiDAR mérések

A vizsgálatokhoz egy Székesfehérvár közelében, Iszkaszentgyörgy település területén 2008 májusában a TELECOPTER cég által készített LiDAR mérések adatait használtam. A LiDAR mérésekkel egy időben a területről ortofotó is készült, a 7.5.1 ábrán ezt használom háttérként. A LiDAR mérések a E18.2280-18.3085 N47.2130-47.2495 földrajzi koordinátákkal megadható, kelet-nyugati irányban körülbelül 6 kilométer, észak-dél irányban körülbelül 4 kilométer kiterjedésű területről készültek. A repülési magasság a mérés során 1400 méter volt.

A vizsgálatokhoz, a bemutatott algoritmus és az összehasonlításként alkalmazott szoftverek esetében is, az utolsó visszaverődésből (last echo) származó pontokat használtam. A mérések legnagyobb magassági hibájára a felmérést végző 15 centimétert adott meg[2].

7.5.2. Geodéziai mérések a tesztterületen

A módszer megbízhatóságának és az eredmény pontosságának a vizsgálata céljából geodéziai módszerekkel is pontokat mértünk fel a LiDAR mérések során bemutatott terület egy részén. A mérések során a terepfelszín 195 pontját mértük fel úgy, hogy változatos felszínborítottságú részeken legyenek. A pontok egy körülbelül 25 hektáros területen helyezkedtek el a N47.2370 E18.2560 pont környezetében.

A mérésekhez Leica TC 407 mérőállomást és Sokkia Stratus geodéziai GNSS vevőt használtunk. A pontok mindegyikét a mérőállomás segítségével, poláris méréssel határoztuk meg. A GNSS vevőt az álláspontok helyzetének statikus módon történő meghatározására használtuk. A méréseket 2015. augusztus 21-én végeztük.

[2]Az adatok a mérés műszaki leírásából származnak.

A geodéziai mérések eredményét a későbbi vizsgálatokban hibátlannak fogadtuk el. A felmért pontok vízszintes helyzetében a terepfelszín magasságának a felmért pont magasságát tekintettük, és mindent ehhez viszonyítottunk.

7.5.3. A LiDAR mérések feldolgozása más eszközökkel

A méréseket a GRASS[3] program [101, 102] segítségével is feldolgoztam. Ennek során első lépésben a `v.outlier` paranccsal szűrtem a pontfelhőt. A következő lépésben a `v.lidar.edgedetection` paranccsal éleket kerestem a pontfelhőben, majd ennek segítségével a `v.lidar.growing` és a `v.lidar.correction` parancsokat használva szűrtem tovább a pontfelhőt. Végül a `v.surf.bspline` paranccsal illesztettem felületet a szűrt felhőre[4]. A GRASS program segítségével készítettem egy szűrés nélküli felületet is. Ehhez a `v.surf.bspline` parancsot közvetlenül az eredeti pontfelhőn alkalmaztam.

A LiDAR méréseket korábban a TopoSys[5] szoftverrel is feldolgozták ennek az eredményeit a pontfelhővel együtt készen vettem át. Az egyik ilyen adat a DTM (Digital Terrain Model) nevet viselő állomány volt, ami a TopoSys program által az utolsó visszaverődésekből számított domborzatmodellt tartalmazta. Ez az állomány üres értékeket tartalmazott azokon a helyeken, ahol a terep felszínét a program nem tudta elég megbízhatóan megállapítani. Az FDTM (Filled DTM) nevű állományban ezeket a helyeket a környező ismert felületdarabokból kiindulva kitöltötték.

7.5.4. A vizsgálatok eredménye

A különféle módon számított felszínmodellek alapján meghatároztam a magasságot a geodéziai módszerrel felmért pontok vízszintes pozíciójában bilineáris interpoláció alkalmazásával (ezt a továbbiakban H_{DEM}-el jelölöm), majd összehasonlítottam ezt a geodéziai módszerrel meghatározott magasságokkal (H_{GEOD}). Az eltéréseknek ($H_{DEM} - H_{GEOD}$) számítottam az átlagát, a mediánját és a szórását. Számítottam továbbá az átlagos abszolút eltérést ($\frac{\sum |H_{DEM} - H_{GEOD}|}{n}$) és az eltérések négyzetes átlagát ($\sqrt{\frac{\sum (H_{DEM} - H_{GEOD})^2}{n}}$) is.

A fenti vizsgálatok elvégzése után a 7.5.1 táblázatban található eredményeket kaptam.

A TopoSys DTM modelljénél a pontok egy része olyan helyre esett, ami a modellen üres volt, így ott csak 158 pontban tudtam megvizsgálni az eltéréseket. A FDTM-nél és a GRASS-os modelleknél már mind a 195 pont esetében össze lehetett hasonlítani a kérdéses felületmodellből kapott magasságot a geodéziai méréssel meghatározottakkal.

A bemutatott módszer vizsgálatakor a magasságokat közvetlenül a vizsgált pozíciókban számoltam ki, majd ezt hasonlítottam össze a geodéziai módszerrel meghatározott magasságokkal. Mivel a módszernek két változtatható paramétere is van, ezt a vizsgálatot sokféle kombinációban elvégeztem.

[3] A program honlapja a https://grass.osgeo.org/ címen található.

[4] A LiDAR adatok GRASS programmal történő feldolgozását a http://www.lutraconsulting.co.uk/blog/2015/04/15/filter-lidar-in-grass/ címen található leírás alapján végeztem.

[5] A programról a http://www.imagemaps.com/toposys.htm oldalon találhatunk információkat.

7.5.1. táblázat. A GRASS illetve a TopoSys programokkal különféle módokon a LiDAR pontfelhőből levezetett felületmodellek magasságainak összevetése a geodéziai mérések eredményeivel.

	GRASS		TopoSys	
	eredeti	szűrt	DTM	FDTM
felhasznált pontok [db]	195	195	158	195
átlag [m]	0,790	0,370	0,051	0,073
medián [m]	0,138	0,103	0,057	0,069
szórás [m]	1,379	0,827	0,162	0,150
átlagos abszolút eltérés [m]	0,790	0,395	0,101	0,104
az eltérések négyzetes átlaga [m]	1,586	0,904	0,169	0,166

7.5.2. táblázat. A különféle R és q paraméterekkel a LiDAR pontfelhőből számolt, és a geodéziai mérésekkel meghatározott magasságok eltéréseinek átlagai. A táblázatban található értékek méterben értendőek.

R [m] / q	0,0055	0,0093	0,016	0,027	0,046	0,079	0,135	0,23	0,393
2,00	0,070	0,070	0,074	0,102	0,133	0,206	0,294	0,418	0,620
2,83	-0,013	-0,002	0,011	0,060	0,086	0,147	0,224	0,373	0,624
4,00	-0,067	-0,064	-0,022	-0,001	0,067	0,122	0,197	0,336	0,619
5,66	-0,075	-0,060	-0,056	-0,038	0,017	0,089	0,182	0,325	0,587
8,00	-0,096	-0,087	-0,069	-0,049	-0,013	0,045	0,144	0,283	0,543
11,31	-0,148	-0,135	-0,126	-0,104	-0,061	-0,008	0,086	0,239	0,507
16,00	-0,219	-0,203	-0,179	-0,156	-0,117	-0,058	0,028	0,170	0,482
22,63	-0,306	-0,279	-0,247	-0,211	-0,172	-0,117	-0,028	0,104	0,440
32,00	-0,425	-0,389	-0,345	-0,304	-0,259	-0,197	-0,120	0,008	0,409
45,25	-0,592	-0,549	-0,506	-0,455	-0,391	-0,313	-0,224	-0,091	0,254

A vizsgálatok során mindkét paraméter értékeit egy mértani sorozat alapján választottam meg. A sugár esetében a sorozat kezdőeleme az 1 méter volt, a szorzó pedig $\sqrt[8]{2}$; az értékeket centiméterre kerekítettem, a legnagyobb vizsgált sugár az $R = 49,35$ volt. A q paraméter sorozatának kezdőértéke és szorzója egyaránt $0,875$ volt; az értékeket négy tizedesre kerekítettem, a legkisebb érték a $q = 0,0048$ volt.

A vizsgálatok eredménye a fentiekből adódóan egy 40 oszlopból (q értékek) és 46 sorból (R értékek) álló táblázatban adható meg. Terjedelmi okokból a dolgozatban található táblázatok (7.5.2, 7.5.3, 7.5.4, 7.5.5, 7.5.6) ezeknek az adatoknak csak egy részét tartalmazzák.

Az 7.5.2 táblázatban megfigyelhetjük, hogy a magasabb q értékeknél az eltérések átlaga pozitív, míg alacsonyabb értékeknél negatív tartományba fordul. Ez logikusan következik a módszer működési elvéből, hiszen minél kisebb hányada kerül a pontoknak a sík alá, a sík annál alacsonyabbra kerül. Az 7.5.3 táblázatban a mediánoknál is az átlagoknál látott trend figyelhető meg.

A szórások értékei (7.5.4 táblázat) jellemzően az alacsonyabb q értékek esetén kisebbek. Hogy a minimum pontosan hol található, az az R értékétől is függ. Szintén a szórásoknál megfigyelt trendeket látjuk az abszolút értékek (7.5.5 táblázat) és a négyzetes

7.5.3. táblázat. A különféle R és q paraméterekkel a LiDAR pontfelhőből számolt, és a geodéziai mérésekkel meghatározott magasságok eltéréseinek mediánjai. A táblázatban található értékek méterben értendőek.

R [m] / q	0,0055	0,0093	0,016	0,027	0,046	0,079	0,135	0,23	0,393
2,00	-0,010	-0,008	-0,004	0,004	0,013	0,027	0,045	0,061	0,089
2,83	-0,031	-0,025	-0,010	-0,001	0,012	0,026	0,037	0,061	0,094
4,00	-0,045	-0,039	-0,022	-0,010	-0,002	0,019	0,036	0,056	0,091
5,66	-0,050	-0,035	-0,029	-0,017	-0,005	0,008	0,029	0,053	0,092
8,00	-0,070	-0,056	-0,049	-0,042	-0,021	-0,006	0,010	0,052	0,102
11,31	-0,102	-0,091	-0,081	-0,066	-0,045	-0,026	0,001	0,045	0,099
16,00	-0,142	-0,126	-0,111	-0,098	-0,080	-0,045	-0,011	0,040	0,101
22,63	-0,217	-0,192	-0,165	-0,126	-0,099	-0,071	-0,023	0,032	0,103
32,00	-0,314	-0,279	-0,240	-0,193	-0,163	-0,103	-0,040	0,030	0,117
45,25	-0,504	-0,431	-0,369	-0,316	-0,230	-0,186	-0,097	0,021	0,151

7.5.4. táblázat. A különféle R és q paraméterekkel a LiDAR pontfelhőből számolt, és a geodéziai mérésekkel meghatározott magasságok eltéréseinek szórásai. A táblázatban található értékek méterben értendőek.

R [m] / q	0,0055	0,0093	0,016	0,027	0,046	0,079	0,135	0,23	0,393
2,00	0,661	0,660	0,663	0,660	0,691	0,824	0,926	1,055	1,329
2,83	0,322	0,320	0,323	0,604	0,623	0,676	0,787	0,961	1,251
4,00	0,221	0,226	0,301	0,353	0,616	0,675	0,761	0,916	1,224
5,66	0,178	0,180	0,295	0,320	0,372	0,611	0,727	0,883	1,159
8,00	0,180	0,200	0,200	0,225	0,276	0,420	0,672	0,810	1,092
11,31	0,206	0,207	0,248	0,245	0,256	0,286	0,482	0,704	1,015
16,00	0,259	0,253	0,246	0,239	0,225	0,296	0,408	0,582	0,949
22,63	0,317	0,291	0,272	0,259	0,249	0,265	0,359	0,508	0,884
32,00	0,388	0,371	0,352	0,338	0,325	0,310	0,313	0,390	0,803
45,25	0,502	0,493	0,486	0,474	0,456	0,442	0,425	0,445	0,684

7.5.5. táblázat. A különféle R és q paraméterekkel a LiDAR pontfelhőből számolt, és a geodéziai mérésekkel meghatározott magasságok eltéréseinek átlagos abszolút értékei. A táblázatban található értékek méterben értendőek.

R [m] / q	0,0055	0,0093	0,016	0,027	0,046	0,079	0,135	0,23	0,393
2,00	0,172	0,172	0,175	0,180	0,199	0,253	0,329	0,440	0,632
2,83	0,114	0,113	0,116	0,151	0,163	0,210	0,274	0,408	0,642
4,00	0,105	0,107	0,109	0,121	0,159	0,194	0,254	0,380	0,644
5,66	0,105	0,100	0,115	0,129	0,146	0,181	0,247	0,373	0,617
8,00	0,117	0,117	0,113	0,120	0,132	0,177	0,237	0,340	0,580
11,31	0,160	0,153	0,155	0,149	0,148	0,151	0,199	0,316	0,555
16,00	0,227	0,213	0,198	0,186	0,167	0,170	0,195	0,280	0,550
22,63	0,311	0,286	0,260	0,232	0,208	0,195	0,203	0,265	0,543
32,00	0,429	0,396	0,354	0,320	0,288	0,251	0,228	0,266	0,571
45,25	0,596	0,554	0,513	0,468	0,415	0,367	0,328	0,324	0,520

7.5.6. táblázat. A különféle R és q paraméterekkel a LiDAR pontfelhőből számolt, és a geodéziai mérésekkel meghatározott magasságok eltéréseinek négyzetes átlagai. A táblázatban található értékek méterben értendőek.

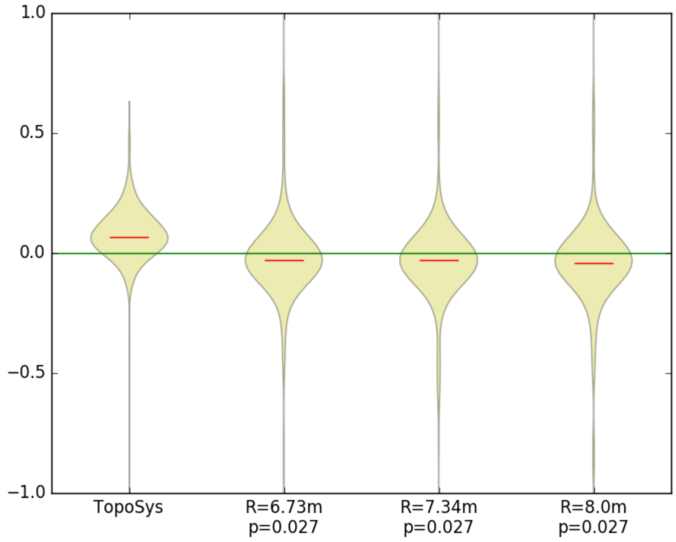
R [m] / q	0,0055	0,0093	0,016	0,027	0,046	0,079	0,135	0,23	0,393
2,00	0,663	0,662	0,665	0,666	0,702	0,847	0,969	1,133	1,463
2,83	0,321	0,320	0,322	0,606	0,628	0,691	0,816	1,029	1,395
4,00	0,231	0,235	0,301	0,352	0,618	0,684	0,784	0,973	1,369
5,66	0,193	0,189	0,300	0,321	0,372	0,616	0,748	0,938	1,296
8,00	0,203	0,218	0,211	0,230	0,276	0,422	0,686	0,856	1,217
11,31	0,253	0,247	0,277	0,266	0,263	0,286	0,488	0,742	1,132
16,00	0,339	0,324	0,304	0,285	0,253	0,301	0,407	0,604	1,062
22,63	0,440	0,403	0,367	0,334	0,302	0,289	0,360	0,518	0,985
32,00	0,575	0,537	0,492	0,454	0,415	0,366	0,334	0,389	0,899
45,25	0,775	0,737	0,701	0,656	0,600	0,541	0,480	0,453	0,728

átlagok (7.5.6 táblázat) esetében is.

Sokféle adat tesztelésével a négyzetes eltérésre a legkedvezőbb értéket az $R = 3,67$ és $q = 0,015$ paraméterek mellett kaptam 0,174 méterrel, ami alig marad el a TopoSys program FDTM modelljétől (0,166) annak ellenére sem, hogy a pontfelhőn előzetesen semmiféle szűrési műveletet nem alkalmaztunk. Az átlagos eltérések és az eltérések mediánja szempontjából a módszer jelentősen jobb eredményeket tudott adni, mint amit a TopoSys FDTM modelljéből kaptunk. Az eloszlások szemléltetése érdekében Violin-plot típusú [72] diagramot is készítettem. (7.5.2. ábra)

Az adatokat vizsgálva megfigyelhető, hogy a szórás és a hozzá hasonlóan viselkedő egyéb jellemzők értéke jelentősen függ a pont környezetének növényzetétől. Ezt egy olyan mérőszámmal fejezhetjük ki amit a $q = 0,95$ és a $q = 0,053$ paraméterekkel $R = 4m$ mellett meghatározott magasságok különbségéből kapunk, és a pontfelhő vastagságának is nevezhetünk.

A korábban bemutatott mérőszámokat külön-külön is kiszámíthatjuk a 40 centiméternél vékonyabb, illetve a 40 centiméternél vastagabb részeken. A 40 centiméternél vé-



7.5.2. ábra. A TopoSys FDTM modell összehasonlítása néhány különféle R és q paraméterrel kapott modellel. A geodéziai módszerekkel felmért pontokban tapasztalt eltérések eloszlásai a ± 1 méteres tartományban. A piros vonalak a mediánokat jelölik. A diagram Matplotlib [75] segítségével készült [6].

konyabb részekre 98, a 40 centiméternél vastagabb részekre 97 pont esett. A DTM modell esetében a 40 centiméternél vékonyabb területekről az összes pontot fel tudtam használni a vizsgálatokhoz, a 40 centiméternél vastagabb helyeken viszont már csak 60 pontban állt rendelkezésre adat. A többi modellnél minden pontban el tudtam végezni az összehasonlítást.

A 7.5.7 adataiból látszik, hogy megfelelően megválasztott paraméterekkel a Disc Fitting eljárással más eljárásokénál jobb vagy azt megközelítő eredményeket lehet elérni. Különösen jónak bizonyul a módszer az átlagok tekintetében.

A módszer hatékonyságának kulcseleme a paraméterek megfelelő megválasztása. Itt megfigyelhető, hogy a kisebb q értékek a hatékonyak, viszont az alacsony arány eléréséhez nagyobb sugárra (R) van szükség, ami a domborzat apróbb részleteinek kinyerését megakadályozza.

Az algoritmust eddig csak Python nyelven [133, 99, 103] implementáltam, mert így a kutatás közben felmerült újabb és újabb ötletek egyszerűen és gyorsan megvalósíthatóak voltak. A végleges algoritmust egy alacsonyabb szintű programozási nyelven (például C-ben vagy C++-ban) implementálva a futásidőt és a memóriahasználatot tekintve jóval hatékonyabb alkalmazást lehetne létrehozni. A módszer egyik előnyös tulajdonsága, hogy nagyon jól párhuzamosítható, mivel a síkok illesztését a létrehozandó domborzatmodell tetszőleges számú pontjában végezhetjük egymással egy időben, közös pontfelhő adatokra támaszkodva, de azokat nem módosítva.

7.5.7. táblázat. A GRASS illetve a TopoSys programokban különféle módszerekkel, illetve a Disc Fitting algoritmussal különféle R és q paraméterek mellett kapott magasságok összehasonlítása a geodéziai mérések eredményével, külön kezelve a 40 centiméternél vékonyabb illetve vastagabb pontfelhőjű területeket. A táblázatban található értékek méterben értendőek.

módszer		GRASS		TopoSys		Disc Fitting				
						$R = 2m$	$R = 4m$		$R = 5,19m$	$R = 10,37$
vastagság		alap	szűrt	DTM	FDTM	$q = 0,09$		$p = 0,016$	$p = 0,014$	$p = 0,0107$
< 40 cm	átlag	0,058	0,060	0,042	0,055	0,005	-0,006	-0,036	-0,054	-0,117
	medián	0,062	0,061	0,048	0,048	0,002	-0,007	-0,037	-0,047	-0,094
	szórás	0,079	0,070	0,158	0,070	0,055	0,056	0,061	0,080	0,145
	átl. absz. elt.	0,080	0,076	0,085	0,072	0,044	0,045	0,055	0,071	0,131
	elt. négyz. átl.	0,097	0,092	0,162	0,162	0,055	0,056	0,070	0,097	0,186
> 40 cm	átlag	1,467	0,683	0,064	0,091	0,471	0,281	-0,007	-0,039	-0,117
	medián	0,858	0,371	0,090	0,094	0,071	0,054	-0,003	-0,012	-0,062
	szórás	1,710	1,086	0,170	0,199	1,154	0,966	0,422	0,229	0,253
	átl. absz. elt.	1,507	0,718	0,126	0,138	0,517	0,366	0,164	0,116	0,151
	elt. négyz. átl.	2,247	1,278	0,181	0,218	1,240	1,002	0,420	0,231	0,277

7.6. További kutatási irányok

Az előző fejezetben bemutatott vizsgálatokból is egyértelműen kiderül, hogy a módszer hatékony alkalmazásának kulcsa az R és q paraméterek megfelelő megválasztása. A vizsgálatok során szintén kiderült, hogy különféle jellegű területeken más és más paraméterek használata lehet célravezető. A javasolt eljárás jövőbeli eredményes, az elterjedt LiDAR feldolgozó programokkal is versenyképes alkalmazásának kulcseleme, hogy kidolgozzunk egy jó becslést az optimális paraméterekre, amelyek akár pontonként eltérőek is lehetnek a terepfelszín adottságainak megfelelően.

Az optimális paraméterek meghatározásakor hasznos lehet a 7.5.7. táblázat kapcsán bemutatott pontfelhő vastagság, amit ott a $q = 0,95$ és a $q = 0,053$ paraméterekkel (mindkét esetben $R = 4m$ sugár mellett) meghatározott síkok távolságaként állítottunk elő. (A pontfelhő vastagság meghatározásának ideális paraméterei szintén érdekes kutatási téma lehet...) Az optimális paraméterek felírhatóak lehetnek tapasztalati függvényekkel a pontfelhő vastagságának függvényeként, ehhez azonban a jelenleginél több referenciamérésre lesz majd szükségünk.

Szintén több referenciamérésre lenne szükség többféle jellegű és elhelyezkedésű mintaterületen a módszer tulajdonságainak részletesebb vizsgálatához.

Nagyon fontos lesz majd megvizsgálni azt is, hogy a bemutatott algoritmus a minél jobb eredmény érdekében hogyan kombinálható más módszerekkel. Itt az eljárás alkalmazása előtt és után is szóba jöhetnek különféle szűrések.

Eddig az eljárás egy előnyös tulajdonságaként, robusztusságának bizonyítékaként mutattam be azt, hogy a feldolgozott pontfelhőn előzőleg nem kellett semmiféle szűrést végezni, bár a last echo adatok felhasználása akár annak is tekinthető. A jövőben érdemes lesz megvizsgálni, hogy az eredmények mennyivel tehetőek pontosabbá, ha előzőleg valamilyen szűrést alkalmazunk a pontfelhőn.

A módszer alkalmazása utáni szűrések lehetőségét is érdemes lesz majd megvizsgálni. Célravezető lehet egyes területeken a magasságot inkább valamilyen interpolációs módszerrel meghatározni a környező, a pontfelhő kiértékelése szempontjából kedvezőbb területek magasságából kiindulva. Az összehasonlító elemzésekben használt TopoSys FDTM modell is ilyen elven működik, az F betű jelentése „filled”, vagyis kitöltött.

7.7. Általánosítási lehetőségek

A bemutatott módszer matematikai hátterét úgy foglalhatjuk össze röviden, hogy hibatűrő módon keresünk egy olyan lineáris függvényt, amivel egy kétdimenziós tér (a két független változó) diszkrét pontjaiban megadott értékeket (a függő változó) jól tudunk közelíteni. Ez az érték az esetünkben a magasság, a kétdimenziós teret pedig a vízszintes koordináták határozzák meg. A diszkrét pontok a lézerszkenneres mérés pontjai, melyeknek vízszintes helyzete és magassága egyaránt ismert; a hibatűrő módon illesztett lineáris függvény pedig az a sík, amivel a terepet szeretnénk a vizsgált pont környezetében közelíteni.

Ezt a feladatot többféle módon is általánosíthatjuk. Sík helyett használhatunk valamilyen magasabb fokú felületet, vagy alkalmazhatjuk a módszert kettő helyett tetszőleges dimenziójú térben is. A LiDAR adatok feldolgozására kigondolt eljárás mindkét esetben az általánosított módszerek egy-egy speciális eseteként lesz felfogható.

7.7.1. Magasabb fokú felületek illesztése

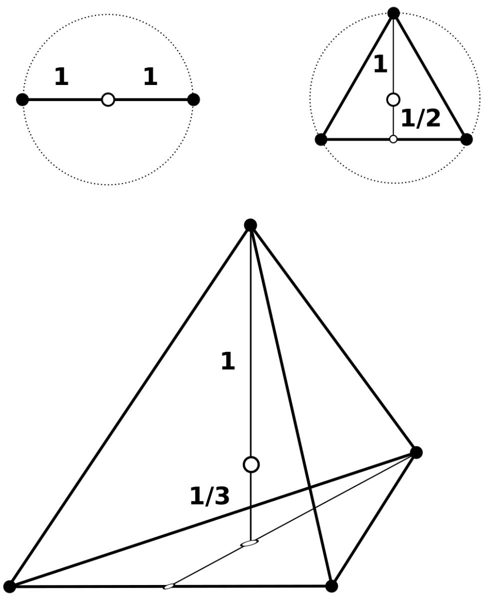
A sík illesztésével kapcsolatban bemutatott elvet tovább lehet fejleszteni magasabb fokú felületek illesztésére is. A vizsgált pozíció körüli, praktikusan kör alakú, területet annyi egyenlő területű szektorra kell felosztani ahány paraméterrel az adott felület megadható, majd feltételként kikötni, hogy valamennyi szektorban a pontok q arányának kell a felület alá esnie.

Az illesztés menetét is a síknál bemutatott elvhez hasonlóan lehetne megoldani. A felületet a szokásos megadási módon (például egy polinom paraméterei a 2.5.7 alapján) túl meg lehet adni a szektorok vezérlőpontjaiban felvett magasságokkal. A két megadási módszer között a kapcsolat megteremthető az egyik irányban az n darab paraméterrel meghatározható felületnek a tér n darab pontjára való illesztésével, a másik irányban pedig az n darab szektorok vezérlőpontjaiban kiszámítva a felület magasságait.

A szektorok mindegyikében meg lehet keresni egymástól függetlenül azt a magasságot, ami alatt a pontok q-ad része található, majd az ezekkel a magasságokkal meghatározott kezdőfelülettel el lehet kezdeni az illesztés műveletét. Az egyes szektorok vezérlőpontjainak magasságait sorban változtatjuk meg úgy, hogy az adott szektorban a pontok q-ad része kerüljön a felület alá. Ezt a műveletet addig kell folytatni, ameddig az összes szektorban a pontok q-ad része lesz a felület alatt, vagyis ameddig n egymást követő lépésben változatlanul hagyjuk a vizsgált szektor súlypontjainak magasságát.

7.7.2. Általánosítás tetszőleges dimenzióra

Az általánosítás egy másik lehetősége az, hogy lineáris függvénynél maradunk, de a tér dimenziószámát tetszőlegesnek vesszük, így kettőnél több vagy akár egy független vál-



7.7.1. ábra. Az egységsugarú gömbben elhelyezkedő N dimenziós tetraéder csúcsainak rekurzív meghatározásához felhasznált összefüggések szemléltetése az $N = 1$, az $N = 2$ és az $N = 3$ esetekben.

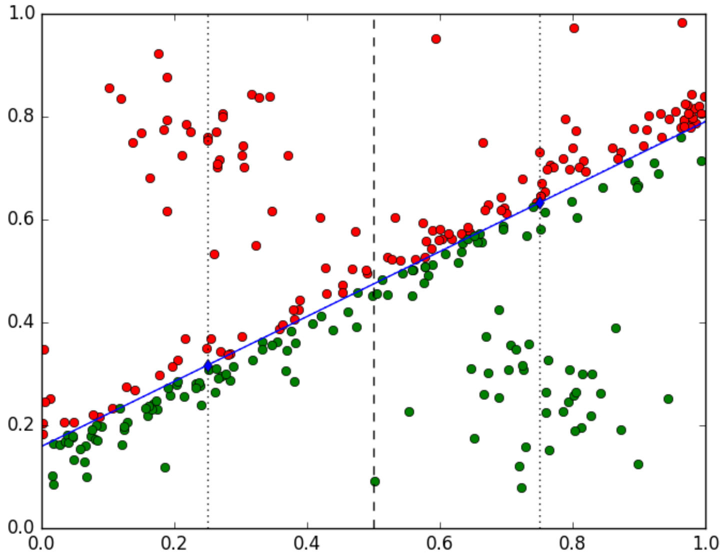
tozóval (7.7.2 ábra) írunk fel egy lineáris regressziót. Az ilyenfajta robusztus illesztésekre már vannak más kidolgozott és publikált módszerek is, például a szélesebb körben használt RANSAC (Random Sample Consensus) [58, 50] vagy a Theil-Sen becslés (Theil-Sen estimator) [129].

Egy N dimenziós skalárteret leíró lineáris függvény $N+1$ paraméterrel rendelkezik, így a módszer általánosított változatában $N+1$ szektorra van szükségünk, amelyeknek az N dimenziós térben kell elhelyezkedniük. Az N dimenziós teret úgy lehet egyszerűen $N+1$ egyforma részre osztani, hogy a szektorok középpontjait egy origó középpontú, N dimenziós tetraéder (egy dimenzióban szakasz, két dimenzióban szabályos háromszög) csúcsaihoz rendelve határozzuk meg; majd minden pontot ahhoz a szektorhoz rendelünk, amelyik középpontjához (az N dimenziós tetraéder $N+1$ csúcsának valamelyike) a legközelebb helyezkedik el.

Az N dimenziós tetraéder csúcsainak koordinátáit egy rekurzív módszerrel határozhatjuk meg. Az első N pontnál az első $N-1$ koordináta az $N-1$ dimenziós tetraéder megfelelő koordinátáinak $\sqrt{1-\frac{1}{N^2}}$ -szerese, az utolsó koordináta pedig $-\frac{1}{N}$. Az utolsó pont utolsó koordinátája 1 az összes többi pedig 0. Az algoritmus rekurzív függvényhívással valósítható meg, mert az N dimenziós tetraéder előállításához az $N-1$ dimenziós tetraéder adataira van szükség. Az $N = 0$ esetben a függvény visszatérési értéke egy olyan lista, ami egyetlen üres listát tartalmaz ([[]]), a rekurzió így itt megáll.

A szektorok középpontjai a fentiek alapján egy origó középpontú és egység sugarú gömbön fognak elhelyezkedni. Ezzel összhangban a vizsgált pozíció R sugarú környezetében elhelyezkedő pontokat is az $x' = \frac{1}{R}(x-p)$ transzformációnak kell alávetni, ahol x' a transzformált, x az eredeti pont, p pedig a vizsgált pozíció helyvektora.

Az algoritmus ezek után a korábbi, kétdimenziós esethez hasonlóan működik. A szek-



7.7.2. ábra. Regressziós egyenes illesztése a módszer egydimenziós változatával. A középső szaggatott vonal osztja a területet két szektorra, a pontvonalak a szektorok középpontjait jelölik. A piros színnel a regressziós egyenes feletti, a zöld színnel pedig a regressziós egyenes alatti pontok vannak jelölve. Ezek száma mindkét szektorban egyforma.

torok középpontjaihoz rendelt kezdeti értékeket a szektor pontjaihoz tartozó értékek mediánjaként (vagy más kvantiliseként) határozzuk meg. Utána szektoronként haladva, az utolsó után az elsővel folytatva, a kérdéses szektor minden pontjára meghatározunk egy lineáris felületet, ami az adott pontra és a többi szektor középpontjára illeszthető, majd ezzel a lineáris felülettel megállapítunk egy értéket a szektor középpontjára; ezeknek az értékeknek a mediánja (vagy más kvantilise) lesz a szektor középpontjához rendelt új érték. Ezt addig ismételjük, amíg egy körben az összes szektor középpontjához rendelt érték változatlan marad. A változatlanság helyett a gyakorlatban érdemes egy határérték (pl. 10^{-5}) alatti változással dolgozni a kerekítési hibák okozta vergődés elkerülése érdekében.

Az általam kidolgozott módszer több szempontból is előnyösebb az eddig publikált megoldásoknál. A futásideje $O\left(mN^3\right)$, ahol m a pontok száma, N pedig a dimenziószám; míg a RANSAC számítási igénye ezzel szemben például exponenciálisan emelkedik a dimenziószám növekedésekor. További előnye az általam kidolgozott módszernek, hogy nem csupán a medián-szerűen (fele felette, fele alatta) tudja illeszteni a felületet, hanem szükség esetén, ha a hibák eloszlása ezt indokolja (mint például a LiDAR mérések esetében), akkor más arány is használható.

A LiDAR mérések kapcsán a most bemutatott általános módszert az $N = 2$ esetben használjuk, a 7.7.2. ábra pedig az $N = 1$ esetet mutatja. A dimenziószám azért tűnik eggyel kevesebbnek a vártnál, mert az egyik koordináta (a magasság) a regresszió függő változója.

A kidolgozott eljárást angolul „Sector Based Linear Regression”-nak, rövidítve SBLR-nek neveztem el.[7] A bemutatott algoritmust egy Python modulként implementáltam.

8. fejezet

Fák modellezése pontfelhők alapján

A lézerszkenneres technológiákat fák felmérésére és modellezésére is fel lehet használni. Az ilyen irányú próbálkozások egy része a fák részletes digitális modelljének létrehozására irányul [82], más részük csupán egyes fontos jellemzőknek, mint például a fák magasságának [81, 104], törzsük átmérőjének [41, 92, 109, 42, 45], leveleik felületének [73], vagy a hasznosítható faanyag térfogatának a meghatározását tűzi ki célul.

Az alábbiakban bemutatom az értekezésem 5. tézisét képező eljárást, ami lehetővé teszi fák ágainak automatizált kiértékelését egy pontfelhő adatai alapján egy viszonylag egyszerű elv segítségével, gömböket illesztve az ágak egyes szakaszaira. A módszer alapelvét a felfelé úszó buborékok ihlették, ezek lesznek az illeszkedő gömbök. Az elképzelt buborék útját a pontfelhőnek az ág felületét leképező pontjai terelgetik a megfelelő irányba. Mivel képzeletbeli buborékunk a helyzete mellett az átmérőjét is változtatni tudja, a modellezett ágat nem csupán egy térbeli vonalláncként állítja elő, hanem meg tudja határozni az ág átmérőjét is az egyes töréspontokban.

Fontos megjegyezni, hogy a következőkben bemutatott módszer csak a nevében hasonlít a „Ball-Pivot Algorithm” vagy röviden BPA névre hallgató eljárásra. [38] Az a módszer pontfelhőre illeszt felületet úgy, hogy egy képzeletbeli gömböt görget a felhő pontjain. A létrejövő felületnek egy-egy csúcspontja lesz az eredményben a felhő minden olyan pontja, amelyikben a gömb megakad. Az általam javasolt módszer teljesen eltér ettől: a pontok összességét vizsgálja (nem csak a gördülő gömbnek az útjába eső pontokat), és az eredménye nem egy általános felület (mesh), hanem egy ág geometriájának a leírása középpontok és átmérők listájaként.

Már jobban hasonlít a bemutatott módszerre a hengerek illesztésén alapuló eljárás. [108] Ettől az általam javasolt módszer abban tér el, hogy henger helyett gömböt használ az illesztéshez, ami a hengernél kevesebb paraméterrel írható le, így kevesebb lehetőséget kell megvizsgálni a kiértékelés során a következő pozíció keresésekor.

8.1. A gömb illesztése

A módszer kulcseleme a gömb illesztésére használt függvény. Ez egy az illeszkedés mértékét jelző számot ad visszatérési értékként, ami azt fejezi ki, hogy a pontfelhőre mennyire illeszkedik egy adott helyzetű és méretű gömb.

8.1.1. A gömb illesztésének alapelve

Minél jobban illeszkedik a gömb a pontfelhőre, a gömb illeszkedését meghatározó függvény annál nagyobb számot ad visszatérési értékként.

$$gömb\,illeszkedés = f\left(pontfelhö, gömb\left(x, y, z, R\right), \left[egyéb\,paraméterek\right]\right) \tag{8.1.1}$$

A gömbnek négy paramétere van: a középpontját három adat segítségével adhatjuk meg (x, y és z), a negyedik paraméter pedig a gömb sugara (R). Az illeszkedés függvényének paramétere természetesen még maga a pontfelhő, és rendelkezhet további paraméterekkel is, amelyek befolyásolhatják az illeszkedés tulajdonságait. Ilyen további paraméter lehet például a módszer érzékenysége a gömbre nem pontosan illeszkedő pontokra.

Többféle függvényt is használhatunk erre a feladatra. A módszerek egyik csoportja (amiket a továbbiakban egyszerű függvényeknek fogok hívni) kizárólag a pontfelhő pontjainak az illeszkedő gömb középpontjától mért távolságait vizsgálja az illeszkedés adatainak meghatározásához. Egy másik csoportba sorolható, a továbbiakban összetett függvényeknek hívott módszerek figyelembe vesznek további, a pontok elhelyezkedésére vonatkozó adatokat is az illeszkedést kifejező szám meghatározásához.

8.1.2. Egyszerű függvények

Az egyszerű függvényeknél definiálunk egy az illeszkedést pontonként vizsgáló függvényt, majd ennek a függvénynek a pontfelhő pontjaira kapott visszatérési értékeit összegezzük.

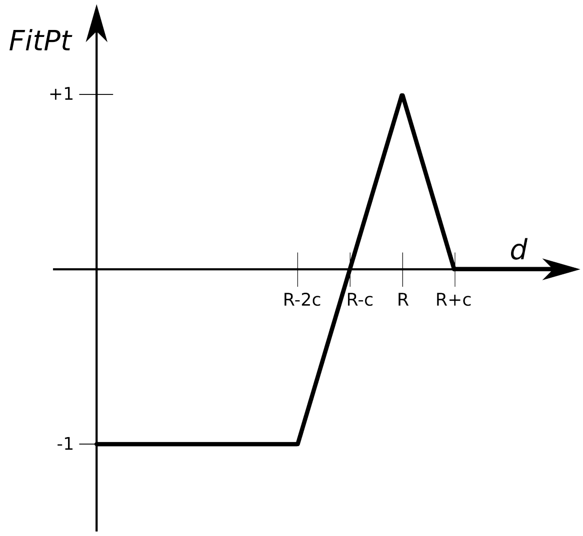
$$gömb\,illeszkedés = \sum pont\,illeszkedés \tag{8.1.2}$$

A pontonkénti illeszkedés vizsgálatához a pont elhelyezkedéséből kizárólag az illeszkedő gömb középpontjától mért távolságát használjuk fel a függvény visszatérési értékének számításához.

$$pont\,illeszkedés = f\left(d, R, \left[egyéb\,paraméterek\right]\right) \tag{8.1.3}$$

Az illeszkedő gömb középpontjától mért távolságon (d) túl még a gömb sugarára R is szükségünk van a pontonkénti illeszkedés mérőszámának meghatározásához. A pont illeszkedését kifejező érték a maximumát akkor veszi fel, amikor a pont a gömb felszínén található, vagyis a d és az R számok egyenlőek. Ha a d értéke jóval nagyobb R-nél, akkor függvénynek nulla értéket kell adnia, hogy a pontfelhő távoli pontjai ne befolyásolják az illesztés eredményét. Az R-nél jelentősen kisebb értékek esetében negatív visszatérési értékeket alkalmazunk, hiszen az ágak belsejében a lézerszkenneres felmérés során nem jöhettek létre pontok. A negatív visszatérési érték egyfajta büntetés a rossz illeszkedések elkerülése érdekében.

A fenti feltételeket sokféle függvény ki tudja elégíteni. Példa egy ilyen függvényre:



8.1.1. ábra. A 8.1.4 szerinti pontilleszkedés függvénye grafikonnal ábrázolva. A d a pont távolsága a gömb középpontjától, az R a gömb sugara, a c pedig az illeszkedés tűrése. A gömb illeszkedésének mérőszáma a pontonkénti értékek összegéből állítható elő.

$$pont\, illeszked\acute{e}s = \begin{cases} -1 & ha\, d \leqq R-2c \\ \frac{d-R+c}{c} & ha\, R-2c < d \leqq R \\ \frac{R-d+c}{c} & ha\, R < d \leqq R+c \\ 0 & ha\, R+c \leqq d \end{cases} \tag{8.1.4}$$

A képletben az R és a d értékek mellett előfordul egy c paraméter is. Ez a konstans szám határozza meg az illeszkedés vizsgálatának tűrését, azt hogy mekkora legyen d-nek az a maximális eltérése az R-től, aminél a függvény még pozitív értéket ad vissza. A függvény a maximális, $+1$ értékét a $d = R$ helyen veszi fel, innen lineárisan esik az $R-c$ és az $R+c$ pontokban visszaadott 0 értékek felé. A $d > R+c$ esetben ez az esés itt megáll, de a $R-2c < d \leqq R$ tartományban még folytatódik a-1 értékig, aminél kevesebb már a $d \leqq R-2c$ tartományban sem lesz. A függvényt grafikonon is ábrázolhatjuk (8.1.1 ábra).

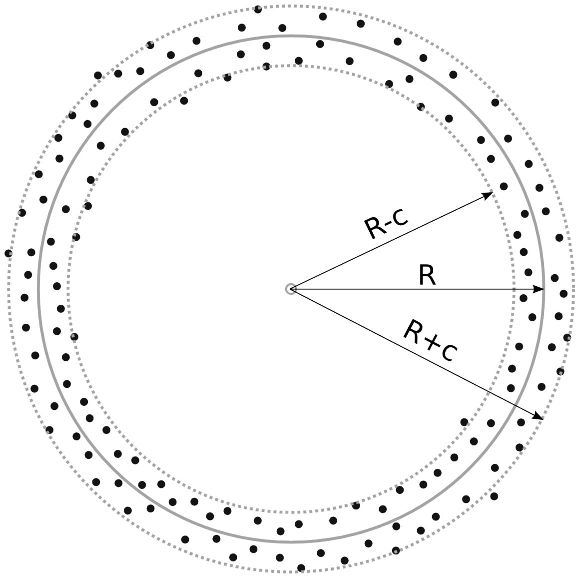
A c értékét az ág keresztmetszetének alakjától (mennyire tér el a körtől) és az ág felületének érdességétől (mennyire tér el a sima felülettől) függően kell felvenni úgy, hogy az ág felszínéhez tartozó pontok pozitív értékeket adjanak.

A bemutatott függvényen túl még számos más megoldás létezhet, hogy az egyszerű illeszkedés során a pont illeszkedésének értékét meghatározzuk. Az 8.1.3 ábra két ilyen lehetséges függvényt is bemutat.

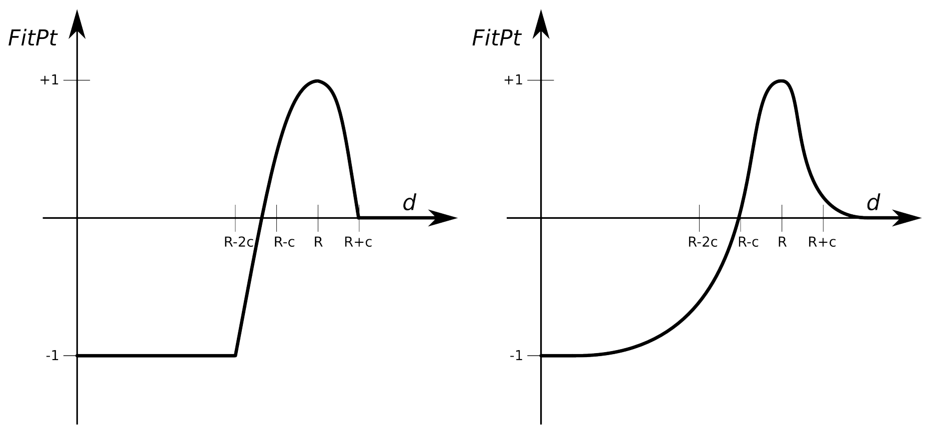
8.1.3. Összetett függvények

Az összetett függvények nem csak egyenként vizsgálják a felhő pontjainak elhelyezkedését, hanem figyelembe veszik azok térbeli eloszlását is. Erre azért lehet szükség, mert enélkül egy sűrűbb felületet csak érintő gömbre is ugyanolyan magas értékeket kaphatunk, mint egy ritkább felületre jól illeszkedőre, amire egy példát a 8.1.4 ábrán láthatunk.

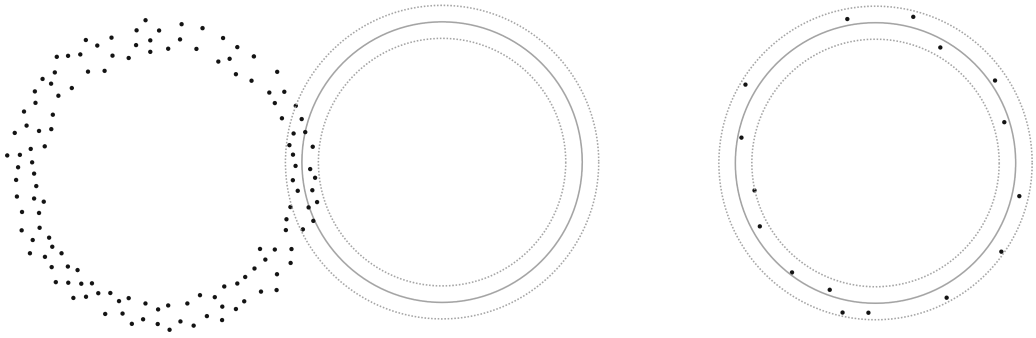
Az illeszkedő pontok eloszlásának figyelembe vételére egy viszonylag egyszerű mód-



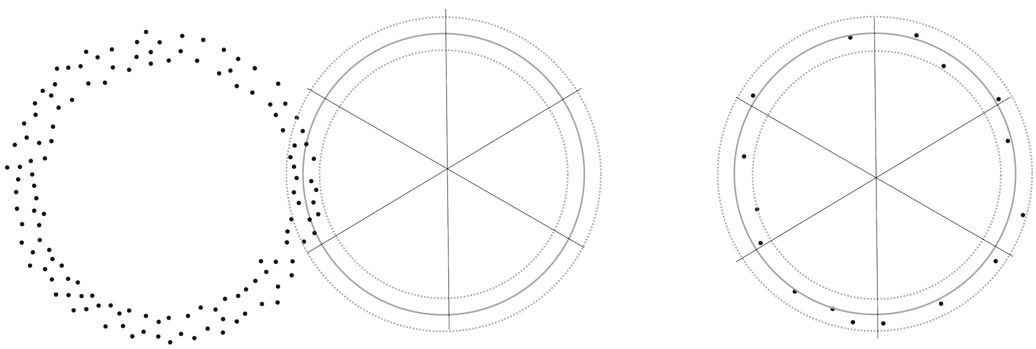
8.1.2. ábra. Az illeszkedő gömb és a pontfelhő pontjainak egy tipikus elhelyezkedése síkban szemléltetve. A folyamatos szürke vonal ábrázolja a gömböt. A szaggatott szürke vonalak határolják azt a területet amely pontjaihoz pozitív illeszkedési értékek tartoznak. A belső szaggatott vonalon belül az illeszkedési érték negatív, a külső szaggatott vonalon kívül pedig nulla.



8.1.3. ábra. Néhány egyéb a pontok illeszkedésének leírására használható függvény. Az alapelv ugyanaz, mint a 8.1.1 ábrán bemutatott függvény esetében: az érték maximuma az $d = R$ helyen van, attól távolodva csökken; kifelé a nulláig, befelé pedig a negatív tartományba is átcsap.



8.1.4. ábra. Egy példa pozitív illeszkedési értéket adó hibás elhelyezkedésű gömbre, és egy vele azonos illeszkedési értéket adó jó illeszkedésre egy ritkább pontfelhő esetén. A hibás illeszkedések egy tipikus esete, amikor a gömb kívülről érintkezik az ággal. Ilyenkor, mivel a gömb belsejébe nem kerülnek pontok, a gömb illeszkedési értéke nem lesz negatív.



8.1.5. ábra. Egy példa összetett illeszkedési függvényre. A korábbiakban megismert pontonkénti illeszkedési értékeket szektoronként összegezve és a legkisebb értékű szektornál kapott eredményt véve végeredménynek elkerülhető az a probléma, hogy egy rosszul illeszkedő gömb a pontfelhő sűrűbb részein jobb eredményt adjon, mini egy jól illeszkedő egy ritka részen.

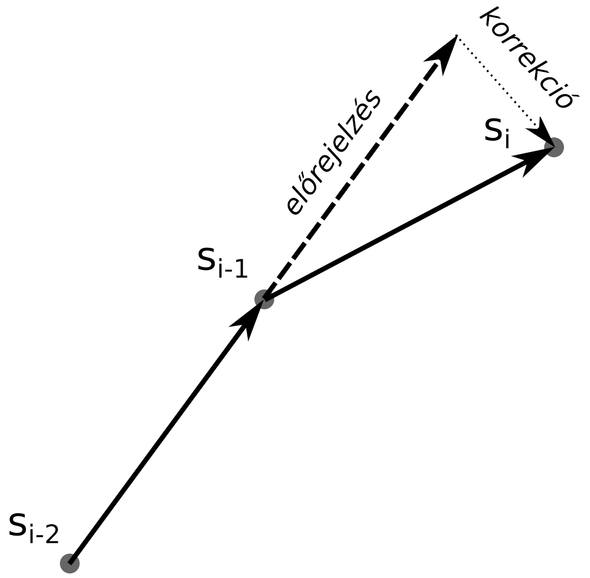
szer lehet, ha gömböt szektorokra osztjuk, és a pontonkénti illeszkedéseket szektoronként összegezzük, majd a gömb illeszkedésének mérőszámaként a szektoronkénti összegek közül a legalacsonyabbat választjuk ki. (8.1.5 ábra)

Meg kell jegyezni, hogy a későbbiekben, az ágak illeszkedő gömbök segítségével történő követése során egymáshoz közeli illeszkedő gömböket hasonlítunk össze, így a pontfelhő sűrűségének változása nem lesz jelentős. A 8.1.5 ábrán bemutatotthoz hasonló módszerekre elsősorban azért van szükség, hogy megakadályozzuk az illeszkedő gömbnek az ág külső felületére kerülését.

8.2. Az ágak követések

8.2.1. Az ágak követésének alapelve

A gömb illesztésével olyan pontokat tudunk keresni, amelyek a fa valamelyik ágának a tengelyvonalának a pontjai lehetnek. A gyakorlatban egy fának vagy annak egy ágának a teljes geometriáját szeretnénk kiértékelni. Ez egymást követő jól illeszkedő gömbök



8.2.1. ábra. Egy ág következő (i indexű) töréspontját meghatározó illeszkedő gömb keresésének alapelve. Az előző és az azt megelőző pozíció közötti vektor lesz az előzőből a keresett pontba mutató vektor előzetes értéke. Ennek a pozíciónak a környékén kell megkeresni a lehető legjobban illeszkedő gömböt. A gömb középpontjának térbeli helyzetén túl ilyenkor a gömb sugarát is változtatjuk. A gömb sugarának előzetes értéke megegyezik az előző illeszkedő gömb sugarával.

sorozatával oldható meg. A gömbök középpontjai megadják az ág alakját, a gömbök átmérőivel pedig az ágnak az egyes helyeken vett átmérőit is megkapjuk. Az ágak keresztmetszetei sokfélék lehetnek [107], de a gömböt általában jól lehet illeszteni minden alakhoz, természetesen a kör keresztmetszethez a legjobban.

Az illeszkedő gömbök sorozatának megkereséséhez szükségünk van egy kezdőpontra és egy kezdőirányra. Ezt követően az ág alakját meghatározó további illeszkedő gömböket már automatikusan meg tudjuk határozni a legjobb soron következő illeszkedés megkeresésével.

A soron következő gömböt az előző gömbtől egy meghatározott távolságra keressük, amit a továbbiakban D-vel fogunk jelölni. Ez a megkötés eggyel csökkenti az illeszkedő gömb elhelyezkedésének szabadságfokát, mivel a középpontja már csak egy két dimenziós felületen (egy D sugarú gömb felszínén) mozoghat. A távolság lehet egy előre megadott fix érték, vagy meghatározható az előző illeszkedő gömb sugarának arányában is, például $D_i = \frac{R_{i-1}}{4}$.

8.2.2. Algoritmus az ágak követésére

A keresést érdemes az előző két illeszkedő gömb által meghatározott irányban, a legelső esetben a megadott kezdőirányban, kezdeni. A keresés során nem csak a gömb középpontjának helyzetét (amit a továbbiakban az $\bar{s}_i$ helyvektorral jelölünk, ahol az alsó indexben szereplő i határozza meg, hogy az ág alakját leíró pozíciósorozat hányadik eleméről van szó) , hanem a sugarát (amit az előzőhöz hasonló elven R_i-vel fogunk jelölni) is változtatni kell.

Az i indexű illeszkedő gömb keresése során tehát egy előrejelzett illeszkedő gömb környezetében érdemes a keresést elkezdeni. Ennek a paramétereit (a középpont koordinátái mellett a gömb sugarát is) az $i-2$ és az $i-1$ indexű illeszkedő gömb paramétereiből lehet megállapítani. Ehhez az előrejelzett pozícióhoz képest a legjobban illeszkedő gömb helyzetét egy korrekciós vektorral tudjuk megadni. (8.2.1 ábra) Az $i-1$ indexű pontból az i indexű pontba mutató előzetes előrejelzett vektort a $\bar{d}'_i = \bar{s}_{i-1} - \bar{s}_{i-2}$ képlettel tudjuk kiszámítani, majd a $\bar{d}_i = \bar{d}'_i \frac{D}{|\bar{d}'_i|}$ összefüggés alkalmazásával gondoskodunk róla, hogy az előrejelzett vektor hossza pontosan a kívánt szakaszhosszúsággal egyezzen meg. (A gömb középpontjának előrejelzett helye ekkor $\bar{s'}_i = \bar{s}_{i-1} + \bar{d}_i$) Hasonló módon számíthatunk egy előrejelzett sugarat is az $R'_i = 2R_{i-1} - R_{i-2}$ módon, vagy vehetjük azt az előző sugárral megegyezőnek: $R'_i = R_{i-1}$.

A korrekciós vektor előállításához első lépésben kiszámítunk két egymásra és az előrejelzett pontba mutató $\bar{d}_i$ vektorra is merőleges vektort (a továbbiakban $\bar{a}_i$-val és $\bar{b}_i$-vel jelöljük őket), melyek hosszát az előző gömb sugarának hányadosaként határozzuk meg: $|\bar{a}_i| = \left|\bar{b}_i\right| = H \cdot R_{i-1}$, ahol H értéke például $0,01$ vagy $0,005$ lehet.

Ehhez először számítani kell az $\bar{a}'_i = \bar{d}_i \times \bar{g}$ vektort, ahol $\bar{g}$ egy tetszőleges, a $\bar{d}_i$-vel nem párhuzamos vektor ($\bar{d}_i \nparallel \bar{g}$). A gyakorlatban $\bar{g}$-nek egy olyan egységvektort lehet alkalmazni, amelyiknek az a koordinátája 1, amelyik koordinátája a $\bar{d}_i$-nek a legkisebb, a többi koordinátája pedig 0. A következő lépésben a $\bar{b}'_i = \bar{d}_i \times \bar{a}'_i$ vektort kell kiszámítani.

A vektoriális szorzat tulajdonságainak köszönhetően a $\bar{d}_i$, az $\bar{a'}_i$ és a $\bar{b'}_i$ vektorok merőlegesek lesznek egymásra. Az $\bar{a}_i = \bar{a'}_i \frac{H}{|\bar{a}'_i|}$ és a $\bar{b}_i = \bar{b'}_i \frac{H}{|\bar{b}'_i|}$ képletekkel számíthatóak a keresés során használni kívánt lépésközzel megegyező hosszúságú vektorok.

Az $\bar{a}_i$ és a $\bar{b}_i$ vektorok egész számú többszöröseinek összegeként áll elő a korrekció vektora: $m_x\bar{a}_i + m_y\bar{b}_i$. A következő illeszkedő gömb középpontjának helyvektora a következőképpen számítható: $s_i = s_{i-1} + d_i + m_x\bar{a}_i + m_y\bar{b}_i$.

Az m_x és m_y számokat a gömb illeszkedését leíró szám maximalizálásával határozzuk meg. A maximális érték meghatározásakor még egy m_R számot is meghatározunk, aminek segítségével az illeszkedő gömb sugarát tudjuk kiszámítani: $R_i = R'_i + H \cdot m_R$.

Az ág követésének algoritmusát a következőképpen foglalhatjuk össze:

Lépés1.0 A kezdeti értékek beállítása: $i = 1$, $\bar{s}_0$ a gömb kezdeti pozíciójának helyvektora, R_0 a gömb kezdeti sugara. $R_{-1} = R_0$ és $\bar{s}_{-1} = \bar{s}_0 - \bar{e}$, ahol $\bar{e}$ a kezdeti irányt meghatározó vektor.

Lépés1.1 Az $F = F_0 = gömbilleszkedés\left(\bar{s}_0,\ R_0\right)$ érték számítása.

Lépés1.2 A D és H paraméterek meghatározása, amennyiben azok nem függenek az előző illeszkedő gömb sugarától.

Lépés2.1 A D és H paraméterek meghatározása, amennyiben azok az előző illeszkedő gömb sugarától függően lettek meghatározva.

Lépés2.2 A $\bar{d}$ vektor számítása: $\bar{d}' = \bar{s}_{i-1} - \bar{s}_{i-2}$, majd $\bar{d} = \bar{d}' \frac{D}{|\bar{d}'|}$.

Lépés2.3 A $\bar{g}$ vektor meghatározása: $\bar{g} = \begin{cases} [1,0,0] & d_x \leqq d_y \, és \, d_x \leqq d_z \\ [0,1,0] & d_y < d_x \, és \, d_y \leqq d_z \\ [0,0,1] & d_z < d_y \, és \, d_z < d_y \end{cases}$

Lépés2.4 Az $\bar{a}' = \bar{d} \times \bar{g}$ és a $\bar{b}' = \bar{d} \times \bar{a}'$ majd az $\bar{a} = \bar{a}' \frac{H}{|\bar{a}'|}$ és a $\bar{b} = \bar{b}' \frac{H}{|\bar{b}'|}$ vektorok számítása.

Lépés2.5 Az $m_x = m_y = m_R = 0$, az R_i' és a $F = gömbilleszkedés\left(\bar{s}_{i-1} + \bar{d},\, R_i'\right)$ kezdeti értékeinek beállítása illetve számítása.

Lépés3.1 Számítsuk az $F_{p,m} = gömbilleszkedés\left(\bar{s}_{i-1} + \bar{d} + (m_x \pm 1)\,\bar{a} + m_y\bar{b},\, R_i' + m_R H\right)$ értékeket. Keressük meg az F, F_p és F_m értékek közül a legnagyobbat. Ha az F_p a legnagyobb, akkor legyen $m_x = m_x + 1$ és $F = F_p$. Ha az F_m a legnagyobb, akkor legyen $m_x = m_x - 1$ és $F = F_m$.

Lépés3.2 Számítsuk az $F_{p,m} = gömbilleszkedés\left(\bar{s}_{i-1} + \bar{d} + m_x\bar{a} + (m_y \pm 1)\,\bar{b},\, R_i' + m_R H\right)$ értékeket. Keressük meg az F, F_p és F_m értékek közül a legnagyobbat. Ha az F_p a legnagyobb, akkor legyen $m_y = m_y + 1$ és $F = F_p$. Ha az F_m a legnagyobb, akkor legyen $m_y = m_y - 1$ és $F = F_m$.

Lépés3.3 Számítsuk az $F_{p,m} = gömbilleszkedés\left(\bar{s}_{i-1} + \bar{d} + m_x\bar{a} + m_y\bar{b},\, R_i' + (m_R \pm 1)\, H\right)$ értékeket. Keressük meg az F, F_p és F_m értékek közül a legnagyobbat. Ha az F_p a legnagyobb, akkor legyen $m_R = m_R + 1$ és $F = F_p$. Ha az F_m a legnagyobb, akkor legyen $m_R = m_R - 1$ és $F = F_m$.

Lépés4 Ha a Lépés3.x során az m_x, m_y vagy az m_R paraméterek bármelyike megváltozott, akkor lépjünk vissza a Lépés3.1-re.

Lépés5 Számítsuk ki az $\bar{s}_i = \bar{s}_{i-1} + \bar{d}_i + m_x\bar{a} + m_y\bar{b}$ és az $R_i = R_i' + m_R H$ értékeket, majd léptessük az i értékét: $i = i + 1$.

Lépés6 Az $F < F_0p$ feltétel és egyéb esetleges befejezési feltételek vizsgálata. Ha teljesül a befejezési feltétel, akkor befejezzük az ág kiértékelését, ha nem akkor lépjünk a Lépés2.1-re.

Az algoritmus lefutása után egy listában a rendelkezésünkre állnak az ág töréspontjainak helyvektorai ($\bar{s}_0,\, \bar{s}_1,\, \bar{s}_2, \ldots,\, \bar{s}_{i-1}$) és az ezekhez tartozó sugarak ($R_0,\, R_1,\, R_2 \ldots,\, R_{i-1}$) adatai. A kiértékelés eredményéből kimarad a listák -1 indexű eleme, mert ez csupán egy fiktív, a Lépés1.0 során számított elem, aminek a szerepe abban van, hogy a későbbi lépésekben, amikor az $i = 1$ indexel kezdődően a soron következő illeszkedéseket keressük, mindig legyen a listának $i - 1$ és $i - 2$ indexű tagja. Kihagyjuk továbbá az utolsó, i értékű indexszel jelölt elemet is, mert arra már teljesült a befejezési feltétel.

Természetesen, ha a -1-es index egy adott implementációban problémát okoz (mert nem lehet nullánál kisebb indexű elem, vagy mert a negatív indexeket a lista végétől kezdődő számozásra használják a kérdéses programozási nyelvben), akkor az elemek indexelését eggyel el kell tolni az itt bemutatotthoz képest. Ekkor az 1-es indexet fogja kapni a kezdőpont, a 0 indexet a kezdőpontot megelőző fiktív pont, a soron következő illeszkedő gömb keresése pedig az $i = 2$ indexel fog kezdődni. Ebben az esetben a keresés eredménye is nyilvánvalóan az $\bar{s}_1,\, \bar{s}_2,\, \bar{s}_3, \ldots,\, \bar{s}_{i-1}$ és az $R_1,\, R_2,\, R_3 \ldots,\, R_{i-1}$ adatok lesznek.

8.2.3. A befejezési feltétel

Az algoritmus működése szempontjából fontos, hogy meg kell határoznunk egy befejezési feltételt, aminek teljesülése a további illeszkedő gömbök keresésének megállítását és az eljárásból való kilépést fogja eredményezni. Ezzel a feltétellel azt kell az algoritmusnak detektálnia, hogy a keresés a kiértékelendő faágnak a végéhez érkezett.

A legegyszerűbb esetben azt figyeljük, hogy a soron következő gömbre kapott illeszkedési érték alacsonyabb-e az F_0-al jelölt kezdeti érték egy bizonyos hányadánál, amit p-vel jelölünk. Például ha a kezdő illeszkedési érték 10 százalékáig akarjuk folytatni a keresést, akkor $p = 0,1$.

Az illeszkedési értéknek a kezdeti érték egy bizonyos hányadosa alá csökkenése mellett érdemes lehet más befejezési feltételeket is meghatározni. A bemutatott módszer alapján elkészített program tesztelése során tapasztaltam, hogy az illeszkedő gömb a keresés során hajlamos mintegy visszapattanni, vagyis miután egy helyen megfordul, az eredeti útvonal irányával ellentétesen is végighaladni az ágon. Ennek az esetnek az elkerülése érdekében befejezési feltételként határozhatjuk meg azt is, ha a következő gömb iránya (az $\bar{s}_i - \bar{s}_{i-1}$ vektor) ellentétes irányú lesz a kezdeti iránnyal (amit korábban $\bar{e}$-vel jelöltünk). Ez a helyzet akkor áll fenn, amikor a két vektor skalárszorzata negatív eredményt ad, vagyis az $\bar{e} \cdot (\bar{s}_i - \bar{s}_{i-1}) < 0$ feltétel igaz.

8.3. A módszer gyakorlati megvalósítása

Az előzőekben bemutatott algoritmus alapján annak a működés közben való tanulmányozására alkalmas programokat készítettem. Ezek közül a legfontosabb az algoritmust is implementáló C++ program volt, aminek az elkészítéséhez a QtCreator[1] 4.7.2 integrált fejlesztő környezetet és a GCC[2] 4.5.2 fordítót használtam.

A program támaszkodik a PCL[3] 1.5.1-es változatára [114]. Ezzel a nyílt forráskódú pontfelhő kezelő programozói könyvtárral oldottam meg a pontfelhők beolvasását és beolvasás utáni tárolását a memóriában egy nyolcasfa (octree) index [97] alapján (a PCL támogatja még a Kd-fa indexet [35] is), valamint a gömb illeszkedések értékeinek számításához szükséges pontgyűjtő lekérdezéseket.

Az egyik fontos, az elkészített program által megvalósított függvény kiszámítja egy gömb (középpont, sugár, valamint az illeszkedés tűrését megadó c paraméter) illeszkedését egy PCL pontfelhőhöz. Ez a függvény egy különálló, más programokból könnyen használható, `sphfitlib` nevű könyvtárba került.

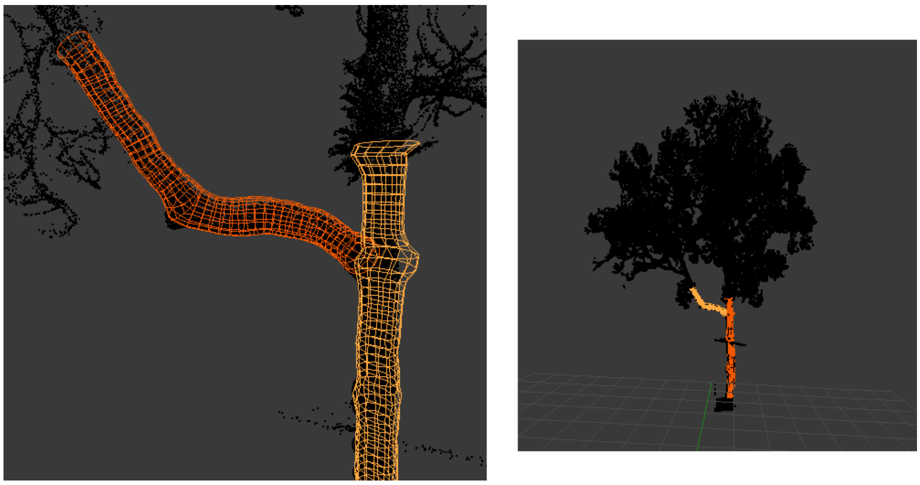
A különféle, az ágak követését megvalósító programok (azért használok többes számot, mert a kutatásaim során többféle megoldást is kipróbáltam) az `sphfitlib` könyvtárra támaszkodva egyszerűen ki tudták számítani egy tetszőleges gömb illeszkedési értékét egy PCL pontfelhőre. Ezeknek a karakteres felületen futtatható programoknak a kimenete egy egyszerű CSV állomány, amelynek az egyes soraiba kerültek az egymás

[1] A program honlapja a http://wiki.qt.io/Qt_Creator/hu címen érhető el.

[2] A GCC egy C, C++ és még több másik programozási nyelv fordítására alkalmas nyílt forráskódú fordítóprogram, neve a GNU Compiler Collection rövidítéséből ered. A program honlapja a https://gcc.gnu.org/ oldalon érhető el.

[3] A PCL (Point Cloud Library) projekt honlapja a http://pointclouds.org/ címen érhető el.

8.3.1. ábra. Az illeszkedő gömbök elhelyezkedése a pontfelhőn (bal oldalt, piros színű körökkel jelölve), és a gömbök pozícióinak és átmérőinak adatsorából generálható felület (jobb oldalt, kék színnel ábrázolva).



8.3.2. ábra. Az illeszkedő gömbök sorozatának adataiból generált felület a gyakorlatban. A felülethálóban minden egyes illeszkedő gömbhöz egy a gömb sugarával arányos méretű szabályos sokszög tartozik. A felületháló az egymást követő sokszögek megfelelő csúcsait összekötve jön létre.

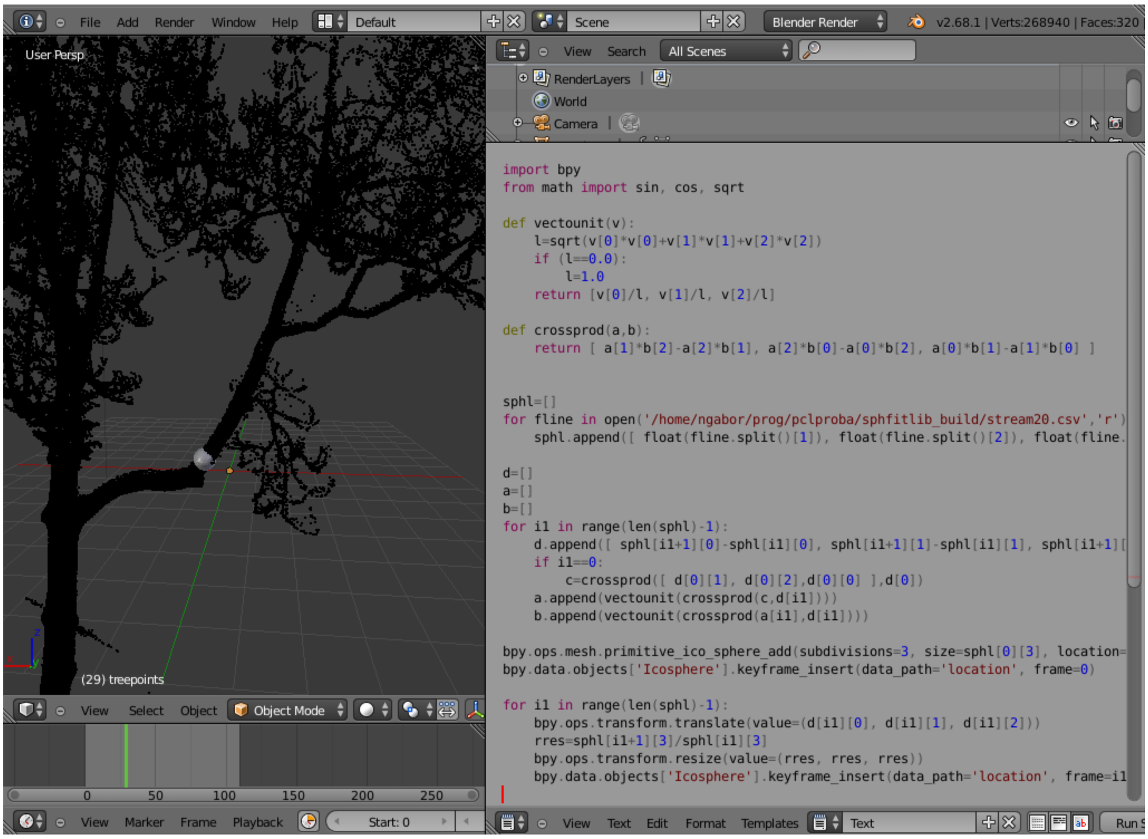
után következő illeszkedő gömbök adatai. (A gömbök középpontjai és sugarai mellett megtalálhatóak ezekben a fájlokban az illeszkedési értékek is.)

A keresés eredményének grafikus megjelenítésére külön programokat készítettem. Ezek közül az egyik poligonhálót generál a kiértékelt ághoz a gömbök középpontjai és sugarai alapján a 8.3.1 ábrán bemutatott elvnek megfelelően, ami így már felületként is jól megjeleníthetővé válik, mint az a 8.3.2 ábrán megtekinthető.

Egy másik program az ágak gömb illesztéssel való kiértékelését egy animáción keresztül teszi szemléletessé. A gömb az animáció egyes képkockáiban a keresés egyes lépéseiben megkapott helyen és sugárral jelenik meg.

Minkét esetben a Blender[4] programot használtam a megjelenítéshez. A CSV állomány adatait az adott megjelenítési formának megfelelő objektumokká (poligonhálóvá, illetve animált gömbbé) alakító programok is részben a Blender alatti alkalmazásfejlesztési le-

[4]A program honlapja a https://www.blender.org/ címen érhető el.



8.3.3. ábra. Az illeszkedő gömb megjelenítése animálva a Blender programban. A jobb oldalon látható annak a Python szkriptnek a forráskódja, amelyik egy CSV állományból beolvasva az ághoz tartozó illeszkedő gömbök adatait, majd hozzáadja az ennek megfelelő animált gömböt a színtérhez. A pontfelhő a színtérben egy csupán csúcsokkal rendelkező mesh objektum. A pontfelhő pontjainak számát csökkentettem a hatékonyabb megjelenítés érdekében.

hetőségek kihasználásával, Python nyelven készültek.

8.4. Továbbfejlesztési lehetőségek

A gyakorlati alkalmazhatóság szempontjából komoly előrelépést jelentene a kiindulási adatok (a kezdő gömb középpontja, sugara és a keresés kezdeti iránya) automatikus megkeresése. Szintén fontos lenne a fák automatizált, a gyakorlatban is jól használható módon történő kiértékelése szempontjából az ágak elágazásainak automatikus detektálására egy jól működő módszert kidolgozni. Ez is alapulhatna a gömb illesztésén, viszont ilyenkor a korábbiaktól lényegesen eltérő irányban és sugárral kellene megtalálni a leágazó elemeket. Az illeszkedő gömbök keresése egy genetikus algoritmussal is megoldható lenne.

A gépi tanulásnak a fák adatainak pontfelhőből történő kiértékelésénél is létjogosultsága van. A [66] például egy a pontfelhőknek az egyes fákhoz tartozó szegmenseit deep learning módszer segítségével leválogató eljárást mutat be. Az illeszkedő gömböket használó módszer is kombinálható lehet deep learning eszközökkel, ha az illeszkedési értékét egy mesterséges neurális hálózaton alapuló regresszióval határozzuk meg. A mesterséges neurális hálózatok térinformatikai alkalmazására többféle hazai példát lehet találni; korábban eredményesen használták már multispektrális felvételek feldolgozására [26], koordináta-transzformációkra [27] és objektumok felismerésére [31] is.

Az illeszkedő gömb ideális paramétereinek meghatározása genetikus algoritmus segítségével is történhet. A gömb illeszkedésének függvénye által adott szám értelemszerűen alkalmazható lenne egy egyed életképességét meghatározó értékként.

A fákról készült pontfelhők feldolgozásának a jövőben komoly szerepe lehet a precíziós mezőgazdaságban is, például gyümölcstermesztésben a metszés automatizálásához elengedhetetlen lesz, hogy a műveletet vezérlő számítógép rendelkezzen a metszendő fa valamilyen térbeli modelljével. [23, 22, 21, 19, 25, 20, 18]

9. fejezet

Összefoglalás

Az értekezésem, annak címével összhangban, a digitális domborzatmodellekkel és pontfelhőkkel kapcsolatos, a terep modellezésében valamilyen formában hasznosítható kutatásaim eredményein alapul. Ezek ismertetése mellett igyekeztem az érintett témákat a terjedelmi korlátokhoz mérten a lehető legjobban bemutatni úgy, hogy a dolgozat egésze minél egységesebb legyen.

A bemutatott kutatásaim során nyert új eredményeket öt darab tézisben foglalhatjuk össze a következőképpen:

1. tézis: Domborzatmodellek tárolásához használható indexelési módszerek kifejlesztése

1/a. altézis: Piramis index alkalmazása szélsőértékekkel GRID típusú domborzatmodellekhez kapcsolódóan

A piramis indexeket elsősorban nagyméretű raszter állományok megjelenítésekor alkalmazzuk azért, hogy a képnek a kisebb felbontásban történő megjelenítése hatékonyabb legyen a külön eltárolt $1/2$, $1/4$, stb. felbontású képeknek köszönhetően, amelyek raszterei ilyenkor a hozzájuk tartozó terület értékeinek átlagát szokták tartalmazni.

Ha egy ilyen piramis indexet egy GRID típusú domborzatmodell esetében (ami ugyanúgy számok kétdimenziós tömbje, mint egy raszter állomány) alkalmazunk úgy, hogy az átlag helyett a legnagyobb és a legkisebb előforduló magasságot tároljuk, akkor az index segítségével az egyes felületdarabok befoglaló téglalapjaihoz jutunk. Ezeket a befoglaló téglalapokat számos a domborzatmodellel kapcsolatban végezhető művelet során felhasználhatjuk.

A 4.1 részben bemutatott témával kapcsolatban 2007-ben [1] és 2016-ban [5] jelentek meg publikációim.

1/b. altézis: TIN domborzatmodellek elemeinek indexelése 2+1 dimenziós R-fa index segítségével

Az R-fa index egy jól bevált és széleskörűen alkalmazott indexelési eljárás többdimenziós térbeli kiterjedéssel rendelkező adatok tárolásakor. Sokféle változata létezik, de mindegyik működésének alapja, hogy a túltelítődő csomópontokat kettévágva illeszti be a szülő csomópontba. A vágás bonyolultsága és számításigénye más tényezők mellett a dimenziószámtól is függ.

A TIN modell elemeit alapesetben egy háromdimenziós R-fa index segítségével lehet indexelni, ez viszont pazarló megoldás, mert a háromszögháló kiterjedése vízszintes irányban nagyságrendekkel nagyobb, mint függőlegesen. Nem lenne viszont célravezető az sem, ha a háromszögháló elemeit csak vízszintes helyzetük alapján, két dimenzióban indexelnénk, mert egyes műveleteknél a befoglaló téglatestre is szükségünk lehet. Az általam javasolt 2+1 dimenziós R-fa index lényege, hogy a csomópontok vágásánál csak a vízszintes helyzetet veszi figyelembe, de ettől függetlenül három dimenzióban számítja a befoglaló téglatesteket. Ezzel a csomópontok vágásának számításigénye is csökken, valamint az így kialakított index szerkezete is kedvezőbb lesz.

A 4.2 részben bemutatott témával kapcsolatban 2007-ben publikációm jelent meg [1].

2. tézis: Egy területen belüli lejtésviszonyok eloszlásának ábrázolása diagram segítségével

Egy terület mezőgazdasági hasznosíthatóságát annak lejtésviszonyai is nagyban befolyásolják, vagyis hogy a terepfelszín milyen irányban és mennyire lejt. Ha egy nagyobb kiterjedésű területről van szó, akkor azon belül különféle lejtésviszonyú részek fordulhatnak elő, melyeknek eloszlását lehetőleg minél szemléletesebben kell ábrázolni.

Az általam javasolt megoldás egy olyan diagramon ábrázolja a területek kitettségét, amelynek alapja egy poláris koordináta-rendszer. Ennek középpontja a vízszintes területeket jelenti, a többi pontja esetében pedig az irány az esésvonal irányával egyezik meg, a távolság pedig a lejtéssel arányos. Ezen a diagramon pontsűrűséggel vagy színfokozatokkal jeleníthető meg a lejtésviszonyok eloszlása. A diagram koordináta-rendszerében akár az egyes növények számára optimális lejtésviszonyokat is lehatárolhatjuk.

A 5. fejezetben bemutatott eredményeimmel kapcsolatban 2016-ban [4] és 2017-ben [3] jelentek meg publikációim.

3. tézis: Terepszerkezeti formák felismerése az irányszög szerinti magasságkülönbségekre illesztett Fourier-sor együtthatóinak vizsgálatával

A terepszerkezeti formák elkülönítésére kidolgozott klasszikus algoritmusok általában azon az elven működnek, hogy a vizsgált ponttól bizonyos távolságban különböző irányokban figyelik a terepfelszín magasságának különbségét a vizsgált ponttól, valamint körbe haladva számolják, hogy ez az érték hányszor vált előjelet. Az általam javasolt módszer az előzőekhez hasonló magasságkülönbségekre az irányszög alapján egy Fourier-sort illeszt, majd ennek paraméterei alapján von le következtetéseket a terepszerkezeti formákra vonatkozóan.

A Fourier-sor paramétereiből kiindulva többféle a terepszerkezeti formákkal kapcsolatos merőszámra tettem javaslatot. A javasolt mérőszámok között vannak olyanok, amelyek fuzzy-alapú elemzésekre alkalmasak (6.3 rész).

A 6.2 részben bemutatott módszer alapjait a [2]-ben ismertettem, további kutatási eredményeim publikálása a témában folyamatban van.

4. tézis: Felületmodell létrehozása pontfelhők alapján az alulról érintő korongok módszerével

Egy (jellemzően LiDAR mérésekből származó) pontfelhőhöz azon az elven rendelünk egy vízszintes (két koordinátával adott) pozícióban egy korongot (síkot), hogy a teret három egyenlő részre osztjuk, majd a síkot úgy helyezzük el, hogy a vizsgált pont adott sugarú környezetében mindhárom harmadban egyenlő, előre meghatározott arányban kerüljenek a sík alá a pontfelhő pontjai.

Ennek az aránynak a medián általánosításaként adódó 50 százaléknál jóval kisebb értéket célszerű felvenni, mert a pontfelhőnek rengeteg pontja képződik a terepfelszín felett (fákon, bokrokon, stb.), ellenben csak mérési hiba folytán kerülhetnek pontok a terepfelszín alá. Ezért hívhatjuk az eljárást alulról érintő korongok módszerének is.

Ezzel a Fitting disc method-nak elnevezett módszerrel nem csupán egy magasságot kapunk, hanem az illesztett sík lejtése alapján a terep lejtésére is rendelkezésre fog állni adat. További lehetőség a pontfelhők eloszlásának vizsgálata, aminek egyik módszere lehet a különféle arányok mellett kapott síkok adatainak összevetése. A javasolt módszerekre programokat fejlesztettem, majd a LiDAR adatok feldolgozásával nyert magassági adatokat geodéziai ellenőrző mérésekkel összevetve vizsgáltam az eljárás megbízhatóságát, amit azután más módszerekkel kapott eredményekkel is összevetettem.

Az 7. fejezetben bemutatott módszert 2017-ben [8] publikáltam.

A LiDAR mérések feldolgozásához kidolgozott módszer matematikai hátterét képező algoritmust általánosítottan tetszőleges dimenziószámú térre. Ennek publikálása szintén folyamatban van. [7]

5. tézis: Fák modellezése pontfelhőkre illeszkedő gömbök módszerével

Egy fáról készült pontfelhő alapján a fa ágainak geometriáját olyan módon értékeljük ki, hogy a fa ágaihoz illeszkedő (a pontfelhőnek a felületük mentén minél több, belsejükben minél kevesebb pontját tartalmazó) gömböket keresünk. Az egymást követő illeszkedő gömbök középpontjai meghatározzák egy ág alakját, átmérőik pedig az ág átmérőjét ez egyes helyeken.

A módszer alapján programot fejlesztettem, a javasolt algoritmust ennek segítségével teszteltem. A 8. fejezetben bemutatott eredményeket 2015-ben publikáltam [9].

A gömbök illeszkedésére kapott mérőszámok a bemutatotton túl sokféle más elemzés alapját is képezhetik.

Publikációk

Téziseket megalapozó publikációk

[1] Nagy Gábor. Digitális domborzatmodellek tárolásának hatékony módszerei. *GEOMATIKAI KÖZLEMÉNYEK*, X.:25–27, 2007.

[2] Nagy Gábor. A terepfelszín mint függvény elemzésének lehetőségei. In J Boda, editor, *Az elmélet és a gyakorlat találkozása a térinformatikában VI.*, pages 285–292. Debreceni Egyetemi Kiadó, 2015.

[3] Nagy Gábor. Egy terület lejtésviszonyainak ábrázolása. In Boglárka Balázs, editor, *Az elmélet és a gyakorlat találkozása a térinformatikában VIII. = Theory meets practice in GIS*, pages 253–258. Debreceni Egyetemi Kiadó, 2017.

[4] G. Nagy. A diagram to illustrate the distribution of slope and aspect of an area. In Gábor Tamás Orosz, editor, *11th International Symposium on Applied Informatics and Related Areas (AIS 2016)*, pages 24–27. Óbudai Egyetem, 2016.

[5] G. Nagy. Analysing the pyramid representation in one arc-second SRTM model. In Gábor Tamás Orosz, editor, *11th International Symposium on Applied Informatics and Related Areas (AIS 2016)*, pages 79–83. Óbudai Egyetem, 2016.

[6] G. Nagy. Matplotlib in the geospatial analyzes. In Gábor Tamás Orosz, editor, *12th International Symposium on Applied Informatics and Related Areas (AIS 2017)*, pages 90–93. Óbudai Egyetem, 2017.

[7] Gábor Nagy. Sector based linear regression, a new robust method for the multiple linear regression. *ACTA CYBERNETICA*, ??(?):???–???, 2018. Megjelenés alatt.

[8] Gábor Nagy, Tamás Jancsó, and Chongcheng Chen. The fitting disc method, a new robust algorithm of the point cloud processing. *ACTA POLYTECHNICA HUNGARICA*, 14(6):59–73, 2017.

[9] Gábor Nagy, Jie Zou, Tamás Jancsó, and Chongcheng Chen. Modeling Tree Branches from Point Clouds by Sphere Fitting Method. In Jancsó Tamás and Engler Péter, editors, *IGIT 2015 International Conference: Integrated Geo-spatial Information Technology and Its Application to Resource and Environmental Management Towards GEOSS*, pages 147–152. Nyugat-magyarországi Egyetem Kiadó, 2015.

Egyéb, az értekezéshez kapcsolódó publikációk

[10] László Bertalan, Csaba Albert Tóth, Gergely Szabó, Gábor Nagy, František Kuda, and Szilárd Szabó. Confirmation of a theory: reconstruction of an alluvial plain development in a flume experiment. *ERDKUNDE*, 70(3):271–285, 2016. UT: 000384466800005 doi: 10.3112/erdkunde.2016.03.05.

[11] Nagy Gábor. *Digitális felületmodell levezetése szintvonalak alapján*. Diplomamunka, Budapesti Műszaki és Gazdaságtudományi Egyetem, 2003.

[12] Nagy Gábor. Korszerű grafikus eszközök lehetőségei a domborzatmodellek térhatású megjelenítésében. *GEOMATIKAI KÖZLEMÉNYEK*, XII.:335–338, 2009.

[13] Nagy Gábor. Lézerszkenneres mérések Székesfehérvár belvárosában. *GEOMATIKAI KÖZLEMÉNYEK*, XIV/1.:149–156, 2011.

[14] M Gede, C Petters, G Nagy, A Nagy, J Mészáros, B Kovács, and C Egri. Laser Scanning Survey in the Pál-völgy Cave, Budapest. In Manfred F Buchroithner, editor, *Proceedings of the 26th International Cartographic Conference*, page 905. International Cartographic Association, 2013.

[15] Mátyás Gede, Zsuzsanna Ungvári, Klaudia Kiss, and Gábor Nagy. Open-source Web-based Viewer Application for TLS Surveys in Caves. In G Gartner and H Haosheng, editors, *Proceedings of the 1st ICA European Symposium on Cartography*, pages 321–328. International Cartographic Association, 2015.

[16] Mátyás Gede, Zsuzsanna Ungvári, and Gábor Nagy. Assessing the accuracy of photogrammetric reconstruction by comparison to laser scanned data. In E Livieratos, editor, *12th Conference on Digital Approaches to Cartographic Heritage*, pages 343–347, 2017.

[17] Gábor Nagy. Interpolation methods for digital elevation models. In Gábor Tamás Orosz, editor, *9th International Symposium on Applied Informatics and Related Areas - AIS2014*, pages 72–74. Óbudai Egyetem, 2014.

[18] Péter Riczu, János Tamás, Attila Nagy, Tünde Fórián, Gábor Nagy, Tamás Jancsó, József Nyéki, et al. 3d laser scanning and modeling of single trees in karcag research center. *Analele Universitatii din Oradea, Fascicula Protectia Mediului*, 17:277–284, 2011.

[19] Riczu P, Csihon Á, Nagy A, Nagy G, Ahmed M E, Tamás J, Gonda I. Intenzív almaültetvény strukturális paramétereinek vizsgálata 3D lézerszkenneres adatok alapján. In Á Ferencz, editor, *Gazdálkodás és Menedzsment Tudományos Konferencia*, pages 198–202. Kecskeméti Főiskola Kertészeti Főiskolai Kar, 2013.

[20] Riczu Péter, Csihon Ádám, Nagy Gábor, Nagy Attilay, Tamás János. Lézerszkenner alapú almadetektálás. *GRADUS*, 3(1):263–267, 2016.

[21] Riczu Péter, Nagy Gábor, Nagy Attila, Tamás János. 3D lézerszkenneres gyomdetektálás gyümölcsültetvényekben. In J P Polgár, editor, *XIX. Ifjúsági Tudományos Fórum*, pages 1x–6x. Pannon Egyetem, Georgikon Kar, 2013.

[22] Riczu Péter, Tamás János, Mesterházi Ákos Péter, Nagy Gábor. Precíziós almatermesztési technológiák fejlesztése a Víz és Környezetgazdálkodási Intézetben. *AGRÁRTUDOMÁNYI KÖZLEMÉNYEK = ACTA AGRARIA DEBRECENIENSIS*, pages 97–103, 2012.

[23] Riczu Péter, Tamás János, Nagy Gábor, Nagy Attila, Fórián Tünde, Jancsó Tamás. A 3D lézerszkenner kertészeti alkalmazhatósága. *AGRÁRTUDOMÁNYI KÖZLEMÉNYEK = ACTA AGRARIA DEBRECENIENSIS*, 46:75–79, 2012.

[24] András Szepes and Gábor Nagy. 3D in GIS. In *RecCAD2010, Alba Iulia*, pages 154–158, 2010.

[25] János Tamás, Péter Riczu, Gábor Nagy, Attila Nagy, Tamás Jancsó, József Nyéki, and Zoltán Szabó. Applicability of 3D laser scanning in precision horticulture. *INTERNATIONAL JOURNAL OF HORTICULTURAL SCIENCE*, 17(4-5):55–58, 2011.

Felhasznált irodalom

[26] Barsi Á. Landsat-felvétel tematikus osztályozása neurális hálózattal. *Geodézia és kartográfia XLIX/4*, pages 21–28, 1997.

[27] Barsi Á. Koordináta-transzformáció megoldása neurális hálózatokkal. *Geodézia és kartográfia LI/10*, pages 12–18, 1999.

[28] Iványi A. *Informatikai algoritmusok.* ELTE Eötvös Kiadó, Budapest, 2004.

[29] Friedrich Ackermann. Digital image correlation: performance and potential application in photogrammetry. *The Photogrammetric Record*, 11(64):429–439, 1984.

[30] V. Balázsik and Z. Tóth. Using geoinformatics tools in archeology. In Gábor Tamás Orosz, editor, *11th International Symposium on Applied Informatics and Related Areas (AIS 2016)*, pages 104–108. Óbudai Egyetem, 2016.

[31] Arpad Barsi. Object detection using neural self-organization. In *Proceedings of the XXth ISPRS Congress, Istanbul, Turkey*, pages 366–371, 2004.

[32] Bruce G Baumgart. Winged edge polyhedron representation. Technical report, DTIC Document, 1972.

[33] Bruce G Baumgart. A polyhedron representation for computer vision. In *Proceedings of the May 19-22, 1975, National Computer Conference and Exposition*, pages 589–596. ACM, 1975.

[34] Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger. The R*-tree: an efficient and robust access method for points and rectangles. In *ACM SIGMOD Record*, volume 19, pages 322–331. Acm, 1990.

[35] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

[36] Berényi Attila, Lovas Tamás, Barsi Árpád. Földi lézerszkenner laboratóriumi vizsgálata. *Geodézia és kartográfia LXII/4*, pages 11–16, 2010.

[37] Berényi Attila, Lovas Tamás, Barsi Árpád, Tóth Zoltán, Rehány Nikolett, Tarsoly Péter. Földi lézerszkennerek minősítő vizsgálatának lehetőségei. *Geomatikai Közlemények XIII.*, pages 87–95, 2010.

[38] Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Claudio Silva, and Gabriel Taubin. The Ball-Pivoting Algorithm for Surface Reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5:349–359, 1999.

[39] Bertalan László, Szabó Gergely. Fotogrammetria-alapú domborzatmodellezés folyóvizes terepasztalon. In B Balázs, editor, *Az elmélet és a gyakorlat találkozása a térinformatikában V.*, pages 69–76. Debreceni Egyetemi Kiadó, 2014.

[40] Atanu Bhattacharya, Manoj K Arora, and Mukat L Sharma. Usefulness of adaptive filtering for improved Digital Elevation Model generation. *Journal of the Geological Society of India*, 82(2):153–161, 2013.

[41] A Bienert, S Scheller, E Keane, F Mohan, and C Nugent. Tree detection and diameter estimations by analysis of forest terrestrial laserscanner point clouds. In *ISPRS workshop on laser scanning*, volume 2007, pages 50–55, 2007.

[42] A Bienert, S Scheller, E Keane, G Mullooly, and F Mohan. Application of terrestrial laser scanners for the determination of forest inventory parameters. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(Part 5), 2006.

[43] James F Blinn. Simulation of wrinkled surfaces. In *ACM SIGGRAPH Computer Graphics*, volume 12, pages 286–292. ACM, 1978.

[44] Laura E Boucheron and Charles D Creusere. Lossless wavelet-based compression of digital elevation maps for fast and efficient search and retrieval. *IEEE transactions on geoscience and remote sensing*, 43(5):1210–1214, 2005.

[45] Gabor Brolly, Geza Kiraly, et al. Algorithms for stem mapping by means of terrestrial laser scanning. *Acta. Silva. Lign. Hung*, 5:119–130, 2009.

[46] Matthew Brown, Richard Szeliski, and Simon Winder. Multi-image matching using multi-scale oriented patches. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 510–517. IEEE, 2005.

[47] Peter Burt and Edward Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on communications*, 31(4):532–540, 1983.

[48] Edwin Catmull and James Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-aided design*, 10(6):350–355, 1978.

[49] L Paul Chew. Constrained delaunay triangulations. *Algorithmica*, 4(1-4):97–108, 1989.

[50] Sunglok Choi, Taemin Kim, and Wonpil Yu. Performance evaluation of ransac family. *Journal of Computer Vision*, 24(3):271–300, 1997.

[51] Paolo Cignoni, Claudio Montani, and Roberto Scopigno. Dewall: A fast divide and conquer delaunay triangulation algorithm in ed. *Computer-Aided Design*, 30(5):333–341, 1998.

[52] Charles D Creusere. Compression of digital elevation maps using nonlinear wavelets. In *Image Processing, 2001. Proceedings. 2001 International Conference on*, volume 3, pages 824–827. IEEE, 2001.

[53] Trevor Darrell and Kwangyoen Wohn. Pyramid based depth from focus. In *Computer Vision and Pattern Recognition, Proceedings CVPR'88.*, pages 504–509. IEEE, 1988.

[54] M Pierrot Deseilligny and I Clery. Apero, an open source bundle adjusment software for automatic calibration and orientation of set of images. In *Proceedings of the ISPRS Symposium, 3DARCH11*, volume 269277, 2011.

[55] Wolfgang Eckstein and Olaf Muenkelt. Extracting objects from digital terrain models. In *SPIE's 1995 International Symposium on Optical Science, Engineering, and Instrumentation*, pages 43–51. International Society for Optics and Photonics, 1995.

[56] N El-Sheimy, C Valeo, and A Habib. Digital terrain modeling-acquisition, manipulation and applications. *Artech House Inc., ISBN 1-58053-921-1*, 2005.

[57] David Eppstein. Beta-skeletons have unbounded dilation. *Computational Geometry*, 23(1):43–52, 2002.

[58] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[59] Peter F Fisher. Extending the applicability of viewsheds in landscape planning. *Photogrammetric Engineering and Remote Sensing*, 62(11):1297–1302, 1996.

[60] Leila De Floriani and Paola Magillo. Visibility algorithms on triangulated digital terrain models. *International Journal of Geographical Information Systems*, 8(1):13–41, 1994.

[61] Janos C Fodor and MR Roubens. *Fuzzy preference modelling and multicriteria decision support.* Springer Science & Business Media, 2013.

[62] Mélykúti Gábor. *Geoinformatika és a digitális felületmodell kapcsolata.* Kandidátusi értekezés, Magyar Tudományos Akadémia, 1994.

[63] Mélykúti Gábor. *Topográfia (moduláris jegyzet).* NymE Geinformatikai Kar, 2011.

[64] K Ruben Gabriel and Robert R Sokal. A new statistical approach to geographic variation analysis. *Systematic Biology*, 18(3):259–278, 1969.

[65] Kristen Grauman and Trevor Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1458–1465. IEEE, 2005.

[66] Haiyan Guan, Yongtao Yu, Zheng Ji, Jonathan Li, and Qi Zhang. Deep learning-based tree classification using mobile LiDAR data. *Remote Sensing Letters*, 6(11):864–873, 2015.

[67] Leonidas J. Guibas, Donald E. Knuth, and Micha Sharir. Randomized incremental construction of delaunay and voronoi diagrams. *Algorithmica*, 7(1):381–413, 1992.

[68] Antonin Guttman. R-trees: A dynamic index structure for spatial searching. ACM 14(2), 1984.

[69] Retter Gyula. *Fuzzy, neurális, genetikus, kaotikus rendszerek: bevezetés a lágy számítás módszereibe.* Akadémiai Kiadó, 2006.

[70] Ralph A Haugerud and DJ Harding. Some algorithms for virtual deforestation (VDF) of LIDAR topographic survey data. *International archives of photogrammetry remote sensing and spatial information sciences*, 34(3/W4):211–218, 2001.

[71] Herman Haverkort, Laura Toma, and Yi Zhuang. Computing visibility on terrains in external memory. *Journal of Experimental Algorithmics (JEA)*, 13:5, 2009.

[72] Jerry L Hintze and Ray D Nelson. Violin plots: a box plot-density trace synergism. *The American Statistician*, 52(2):181–184, 1998.

[73] Ronghai Hu, Guangjian Yan, Françoise Nerry, Yunshu Liu, Yumeng Jiang, Shuren Wang, Yiming Chen, Xihan Mu, Wuming Zhang, and Donghui Xie. Using airborne laser scanner and path length distribution model to quantify clumping effect and estimate leaf area index. *atmosphere*, 1:6, 2018.

[74] David A Huffman et al. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952.

[75] John D. Hunter. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95, May-Jun 2007.

[76] Elek I. Domborzati modellek és a mintavételi tétel (I. rész). *Geodézia és Kartográfia LVI.*, 10:21–24, 2004.

[77] Elek I. Domborzati modellek és a mintavételi tétel (II. rész). *Geodézia és Kartográfia LVI.*, 11:21–24, 2004.

[78] Andy Jarvis, Hannes Isaak Reuter, Andrew Nelson, Edward Guevara, et al. Hole-filled SRTM for the globe Version 4. *available from the CGIAR-CSI SRTM 90m Database (http://srtm. csi. cgiar. org)*, 2008.

[79] Michael Kalbermatten, Dimitri Van De Ville, Pascal Turberg, Devis Tuia, and Stéphane Joost. Multiscale analysis of geomorphological and geological features in high resolution digital elevation models using the wavelet transform. *Geomorphology*, 138(1):352–363, 2012.

[80] Xiaochen Kang, Jiping Liu, and Xiangguo Lin. Streaming progressive tin densification filter for airborne lidar point clouds using multi-core architectures. *Remote Sensing*, 6(8):7212–7232, 2014.

[81] G Király and G Brolly. Tree height estimation methods for terrestrial laser scanning in a forest reserve. *Proceedings of the ISPRS Workshop Laser Scanning*, pages 12–14, 2007.

[82] Géza Király and Gábor Brolly. Modelling single trees from terrestrial laser scanning data in a forest reserve. *Photogrammetric Journal of Finland*, 21(1):37–50, 2008.

[83] Jan Klein and Gabriel Zachmann. Point cloud surfaces using geometric proximity graphs. *Computers & Graphics*, 28(6):839–850, 2004.

[84] Zsófia Koma, Balázs Székely, Zoltán Folly-Ritvay, Ferenc Skobrák, Kristina Koenig, and Bernhard Höfle. Automated identification and geometrical features extraction of individual trees from mobile laser scanning data in budapest. In *EGU General Assembly Conference Abstracts*, volume 18, page 10237, 2016.

[85] K. Kraus and N. Pfeifer. Determination of terrain models in wooded areas with airborne laser scanner data. *{ISPRS} Journal of Photogrammetry and Remote Sensing*, 53(4):193–203, 1998.

[86] Franz Leberl, Arnold Irschara, Thomas Pock, Philipp Meixner, Michael Gruber, Set Scholz, and Alexander Wiechert. Point clouds. *Photogrammetric Engineering & Remote Sensing*, 76(10):1123–1134, 2010.

[87] Der-Tsai Lee and Bruce J Schachter. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer & Information Sciences*, 9(3):219–242, 1980.

[88] Zhilin Li, Christopher Zhu, and Chris Gold. *Digital terrain modeling: principles and methodology.* CRC press, 2004.

[89] Peter Lohmann, Andreas Koch, and Michael Schaeffer. Approaches to the filtering of laser scanner data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1; PART 3):540–547, 2000.

[90] T Lovas, A Barsi, A Detrekoi, L Dunai, Z Csak, A Polgar, A Berenyi, Z Kibedy, and K Szocs. Terrestrial laser scanning in deformation measurements of structures. *International Archives of Photogrammetry and Remote Sensing*, 37(B5):527–531, 2008.

[91] Tamás Lovas, Nikolett Rehány, and József árpád Somogyi. Történelmi épületek rekonstrukciós munkálatainak támogatása pontfelhők segítségével. *Geodézia és kartográfia LXX/1*, pages 19–24, 2018.

[92] J. L. Lovell, D. L. B. Jupp, G. J. Newnham, and D. S. Culvenor. Measuring tree stem diameters using intensity profiles from ground-based scanning lidar from a fixed viewpoint. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(1):46–55, 2011.

[93] Béla Márkus. Terrain analysis in consideration of surface curvature conditions. *Per.Pol.Civil Engineering, Budapest*, 1986/1-2, 1986.

[94] Béla Márkus and T. Molnár. Terrain description by cluster analysis. *BME Newsletter*, 1984/1, 1984.

[95] Paul Martz. *OpenGL distilled*. Addison-Wesley Professional, 2006.

[96] Arne Maus. Delaunay triangulation and the convex hull of n points in expected linear time. *BIT Numerical Mathematics*, 24(2):151–163, 1984.

[97] Donald Meagher. Geometric modeling using octree encoding. *Computer graphics and image processing*, 19(2):129–147, 1982.

[98] Xuelian Meng, Nate Currit, and Kaiguang Zhao. Ground filtering algorithms for airborne LiDAR data: A review of critical issues. *Remote Sensing*, 2(3):833–860, 2010.

[99] K Jarrod Millman and Michael Aivazis. Python for scientists and engineers. *Computing in Science & Engineering*, 13(2):9–12, 2011.

[100] Pierre Moulon, Pascal Monasse, Romuald Perrot, and Renaud Marlet. Openmvg: Open multiple view geometry. In *International Workshop on Reproducible Research in Pattern Recognition*, pages 60–74. Springer, 2016.

[101] Markus Neteler, M Hamish Bowman, Martin Landa, and Markus Metz. Grass gis: A multi-purpose open source gis. *Environmental Modelling & Software*, 31:124–130, 2012.

[102] Markus Neteler and Helena Mitasova. *Open source GIS: a GRASS GIS approach*, volume 689. Springer Science & Business Media, 2013.

[103] Travis E Oliphant. Python for scientific computing. *Computing in Science & Engineering*, 9(3), 2007.

[104] Kenneth Olofsson, Johan Holmgren, and Håkan Olsson. Tree stem and height measurements using terrestrial laser scanning and the ransac algorithm. *Remote sensing*, 6(5):4323–4344, 2014.

[105] Thomas K Peucker, Robert J Fowler, James J Little, and David M Mark. The triangulated irregular network. In *Amer. Soc. Photogrammetry Proc. Digital Terrain Models Symposium*, volume 516, page 532, 1978.

[106] N Pfeifer, T Reiter, C Briese, and W Rieger. Interpolation of high quality ground models from laser scanner data in forested areas. *International Archives of Photogrammetry and Remote Sensing*, 32(3/W14):31–36, 1999.

[107] Norbert Pfeifer and Daniel Winterhalder. Modelling of tree cross sections from terrestrial laser scanning data with free-form curves. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci*, 36(8/W2):76–81, 2004.

[108] Pasi Raumonen, Mikko Kaasalainen, Markku Åkerblom, Sanna Kaasalainen, Harri Kaartinen, Mikko Vastaranta, Markus Holopainen, Mathias Disney, and Philip Lewis. Fast automatic precision tree models from terrestrial laser scanner data. *Remote Sensing*, 5(2):491–520, 2013.

[109] R Suraj Reddy, CS Rakesh, CS Jha, and KS Rajan. Automatic estimation of tree stem attributes using terrestrial laser scanning in central indian dry deciduous forests. *CURRENT SCIENCE*, 114(1):201–206, 2018.

[110] Fabio Remondino, Maria Grazia Spera, Erica Nocerino, Fabio Menna, and Francesco Nex. State of the art in high density image matching. *The Photogrammetric Record*, 29(146):144–166, 2014.

[111] Péter Riczu, Attila Nagy, Éva Lehoczky, and János Tamás. Precision weed detection using terrestrial laser scanning techniques. *Communications in Soil Science and Plant Analysis*, 46(sup1):309–316, 2015.

[112] Ernesto Rodriguez, Charles S Morris, and J Eric Belz. A global assessment of the SRTM performance. *Photogrammetric Engineering & Remote Sensing*, 72(3):249–260, 2006.

[113] Randi J Rost, Bill Licea-Kane, Dan Ginsburg, John M Kessenich, Barthold Lichtenbelt, Hugh Malan, and Mike Weiblen. *OpenGL shading language*. Pearson Education, 2009.

[114] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *International Conference on Robotics and Automation*, Shanghai, China, 2011.

[115] Alessandro Samuel-Rosa, Lúcia HC dos Anjos, Gustavo M Vasques, Mauro AH Antunes, and Ricardo SD Dalmolin. Identifying and correcting oblique striping in the Topodata digital elevation model. In *XXXIV Brazilian Congress of Soil Science*, 2013.

[116] Sárközy Ferenc, Márkus Béla. *Geodéziai AMT*. Tankönyvkiadó, Budapest, 1986.

[117] Mark Segal and Kurt Akeley. The Design of the OpenGL graphics interface. In *Silicon Graphics Computer Systems*. Citeseer, 1994.

[118] Jonathan Richard Shewchuk. Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator. In *Applied computational geometry towards geometric engineering*, pages 203–222. Springer, 1996.

[119] Jonathan Richard Shewchuk. A condition guaranteeing the existence of higher-dimensional constrained Delaunay triangulations. In *Proceedings of the fourteenth annual symposium on Computational geometry*, pages 76–85. ACM, 1998.

[120] Jonathan Richard Shewchuk. Delaunay refinement algorithms for triangular mesh generation. *Computational geometry*, 22(1):21–74, 2002.

[121] David Sinclair. S-hull: a fast radial sweep-hull routine for delaunay triangulation. *arXiv preprint arXiv:1604.01428*, 2016.

[122] George Sithole. Filtering of laser altimetry data using a slope adaptive filter. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/W4):203–210, 2001.

[123] George Sithole and George Vosselman. Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(1-2):85–101, 2004.

[124] Peter Su and Robert L Scot Drysdale. A comparison of sequential delaunay triangulation algorithms. *Computational Geometry*, 7(5-6):361–385, 1997.

[125] Szirmay-Kalos László, Antal György, Csonka Ferenc. *Háromdimenziós grafika, animáció és játékfejlesztés.* ComputerBooks, Budapest, ISBN: 963 618 303 1, 2003.

[126] Tetushi Tachikawa, Manabu Kaku, Akira Iwasaki, Dean B Gesch, Michael J Oimoen, Zheng Zhang, Jeffrey J Danielson, Tabatha Krieger, Bill Curtis, Jeff Haase, et al. ASTER global digital elevation model version 2-summary of validation results. Technical report, NASA, 2011.

[127] Tarsoly Péter. Lézerszkenner alkalmazása gyapjúzsákok és ingókövek felmérésében a Velencei-hegységben. *Karsztfejlődés*, XVIII:155–166, 2013.

[128] Telbisz Tamás, Székely Gábor, Timár Gábor. *Digitális terepmodellek; adat, látvány, elemzés.* ELTE TTK FFI Természetföldrajzi Tanszék, ISBN 978 963 284 372 8, 2013.

[129] Henri Theil. A rank-invariant method of linear and polynomial regression analysis. In *Henri Theil's Contributions to Economics and Econometrics*, pages 345–381. Springer, 1992.

[130] Timár G, Telbisz T, Székely B. Űrtechnológia a digitális domborzati modellezésben: az SRTM adatbázis. *Geodézia és Kartográfia*, 55(12):11–15, 2003.

[131] Charles K Toth, Zoltan Koppanyi, Dorota A Grejner-Brzezinska, and Grzegorz Jóźków. Spatial spectrum analysis of various digital elevation models. In *ASPRS Annual Conference, Louisville, KY*, 2014.

[132] Ungvári Zsuzsanna. *A térképi generalizálás vizsgálata különféle méretaránytartományokban domborzatmodelleken.* PhD értekezés, 2017. Eötvös Loránd Tudományegyetem Földtudományi Doktori Iskola Térképészet Doktori Program.

[133] Guido Van Rossum et al. Python Programming Language. In *USENIX Annual Technical Conference*, volume 41, 2007.

[134] Jakob J Van Zyl. The Shuttle Radar Topography Mission (SRTM): a breakthrough in remote sensing of topography. *Acta Astronautica*, 48(5):559–565, 2001.

[135] Remco C Veltkamp. The γ-neighborhood graph. *Computational Geometry*, 1(4):227–246, 1992.

[136] Remco C Veltkamp. The γ-neighborhood graph. *Closed Object Boundaries from Scattered Points*, pages 23–36, 1994.

[137] George Vosselman. Slope based filtering of laser altimetry data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/2; PART 3):935–942, 2000.

[138] Stéfan van der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

[139] Jianjun Wang, Gary J Robinson, and Kevin White. Generating viewsheds without using sightlines. *Photogrammetric engineering and remote sensing*, 66(1):87–90, 2000.

[140] Frank Warmerdam. The geospatial data abstraction library, open source approaches in spatial data handling. pages 87–104. Springer, 2008.

[141] Joanne C White, Michael A Wulder, Mikko Vastaranta, Nicholas C Coops, Doug Pitt, and Murray Woods. The utility of image-based point clouds for forest inventory: A comparison with airborne laser scanning. *Forests*, 4(3):518–536, 2013.

[142] Bin Wu, Bailang Yu, Chang Huang, Qiusheng Wu, and Jianping Wu. Automated extraction of ground surface along urban roads from mobile laser scanning point clouds. *Remote Sensing Letters*, 7(2):170–179, 2016.

[143] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[144] Keqi Zhang, Shu-Ching Chen, Dean Whitman, Mei-Ling Shyu, Jianhua Yan, and Chengcui Zhang. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *Geoscience and Remote Sensing, IEEE Transactions on*, 41(4):872–882, 2003.

[145] Wuming Zhang, Jianbo Qi, Peng Wan, Hongtao Wang, Donghui Xie, Xiaoyan Wang, and Guangjian Yan. An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation. *Remote Sensing*, 8(6):501, 2016.

[146] Qing Zhu, Deren Li, Yeting Zhang, Zheng Zhong, and Duo Huang. CyberCity GIS (CCGIS): integration of DEMs, images, and 3D models. *Photogrammetric engineering and remote sensing*, 68(4):361–368, 2002.