

Óbudai Egyetem
Neumann János Informatikai Kar
Szoftvertchnológia Intézet



TUDOMÁNYOS DIÁKKÖRI DOLGOZAT

VASTAGBÉLRÁK DIAGNOSZTIKA DIGITÁLIS SZÖVETI MINTÁKON

Szerzők: **Kerekes Zoltán**
mérnök informatikus szak, IV. évf.
Tóth Zoltán
mérnök informatikus szak, IV. évf.

Konzulensek: **Dr. Vámosy Zoltán**
egyetemi docens
Dr. Sergyán Szabolcs
egyetemi docens

Tartalomjegyzék

Bevezetés	4
1. Áttekintés	4
1.1. A vastagbélrák	4
1.2. A digitális szövettan	4
1.3. A vastagbél felépítése	6
2. Hasonló rendszerek bemutatása	7
2.1. Colon	7
2.1.1. A projekt bemutatása	7
2.1.2. Sejtmagok detektálása	8
2.1.3. Mirigyek detektálása	8
2.1.4. Hám detektálás	9
2.1.5. A projekt értékelése	9
2.2. Automatizált rákdiagnosztika hisztopatológiai képek alapján	10
2.2.1. Előfeldolgozás	10
2.2.1.1. Régió alapú sejtdetektálás	10
2.2.1.2. Határkeresésen alapuló sejtdetektálás	10
2.2.2. Jellemzők vizsgálata	11
2.2.2.1. Morfológiai jellemzők	11
2.2.2.2. Texturális jellemzők	11
2.2.2.3. Fraktál-alapú jellemzők	11
2.2.2.4. Topológiai jellemzők	12
2.2.2.5. Intenzitás-alapú jellemzők	12
2.2.2.6. Jellemző kiválasztás	12
2.2.3. Diagnózis	13
2.2.4. A projekt értékelése	14
2.3. Vastagbél mirigyek szegmentálása objektum gráf használatával	14
2.3.1. Áttekintés	14
2.3.2. A projekt értékelése	16
3. Használandó módszerek, algoritmusok, keretrendszerek	17

3.1.	Színterek.....	17
3.2	Szín Struktúra Kód.....	18
3.2.1	Az algoritmus felépítése.....	19
3.2.2	Az algoritmus működése.....	20
3.2.2.1	Inicializálás	20
3.2.2.2	Összekapcsolás.....	20
3.2.2.3	Szétvágás.....	21
3.3	Soros kapcsolódó komponens módszer	21
3.4.	Aforge.Net.....	22
3.5.	OpenCV	22
4.	Saját rendszer bemutatása	23
4.1.	Célkitűzés	23
4.2.	Rendszervázlat	23
4.3	A rendszer működésének bemutatása	24
4.3.1	Mirigydetektáláshoz megvalósított funkciók.....	24
4.3.1.1	Szín Struktúra Kód (Color Structure Code).....	24
4.3.1.2	RGB-ből LAB színtérbe konvertálás.....	24
4.3.1.3	Küszöbölés	24
4.3.1.4	Összekötés	24
4.3.1.5	Összefüggő komponens szegmentáció	25
4.3.1.6	Határvonal létrehozás és illesztés.....	26
4.3.1.7	Hibásan detektált mirigyek eltávolítása.....	28
4.3.1	A sejtmagdetektáláshoz megvalósított funkciók	29
4.3.1.1	Sejtmagdetektálás HSV színtér használatával.....	29
4.3.1.2	Sejtmagdetektálás színszegmentálás alkalmazásával.....	30
4.3	Tesztelés és eredmények bemutatása.....	31
4.3.1	Mirigydetektálás eredményei	31
4.3.1	Az eredmények összehasonlítása más rendszerekkel	37
4.3.2	Sejtmagok detektálásának tesztelése	37
4.4	Továbbfejlesztési lehetőségek	38
	Összegzés, a projekt értékelése	39
	Irodalomjegyzék.....	41

Függelék	43
1 Vastagbél mirigyek szegmentálása objektum gráf használatával – részletes kifejtés	43
1.1 Pixelek osztályozása	43
1.2 Circle-Fit transzformáció	44
1.3 Mirigyjelöltek detektálása.....	46
1.3.1 A hibásan detektált sejtmag felismerése és eltüntetése	46
1.3.2 Mirigymag detektálás.....	47
1.3.3 Régiónövelés	49
1.4 Hibásan detektált mirigyek elkülönítése.....	50
1.5 Mirigyhatárvonalak felismerése	51

Bevezetés

A projekt célja egy olyan szoftver elkészítése, amely megoldást jelenthet a vastagbélrák diagnosztizálásának nehézségeire, ezzel elősegítve a patológusok munkáját és a betegség korai felismerését. Mindez mikroszkópikus szöveti képek feldolgozásával és elemzésével valósul meg. A tanulmány áttekinti a témával kapcsolatos alapvető ismereteket, hasonló projekteket, valamint az általunk tervezett és elkészített rendszert.

1. Áttekintés

1.1. A vastagbélrák

A kolorektális rákokban szenvedők száma évről évre növekvő tendenciát mutat. Ez a szám olyannyira megnőtt, hogy a rákos megbetegedések sorában a második helyet foglalja el a tüdőrák okozta elváltozások után. Magyarországon évente 8 ezer új esetet regisztrálnak, ez azt jelenti, hogy kb. minden huszadik embert érinthet a kór. A vastagbélrákban szenvedők 25–30%-a marad életben 5 évvel a betegség kialakulása után, amelyhez nagymértékben hozzájárul a késői diagnosztizálás, részben a szűrővizsgálatok elhanyagolása miatt. Ennek a kórnak a kialakulása egy hosszú, körülbelül 10, gyakran 15 éves folyamat is lehet. Azoknak a betegeknek, akiknél a betegség korai fázisában megszületik a diagnózis, a kezelési esélyeik sokkal jobbak, mint a késői stádiumú, panaszokkal jelentkezőké. Tehát a megfelelő szűrésekkel a betegség korai stádiumban felismerhető, ami nagymértékben növeli a kezelés hatékonyságát, de általánosságban elmondható, hogy az emberek többsége ezeket sajnos elhanyagolja [MABE09], [DELA08].

1.2. A digitális szövettan

Az általunk írt programmal digitális képként tárolt szövetmintát kívánunk elemezni és ezáltal következtetéseket levonni annak állapotáról. De mik ezek a szövetminták és hogyan állítják őket elő? A szakorvosok biopszia segítségével a testből sejteket vagy szöveteket távolítanak el [PATO12]. A metszetekben lévő szöveteket a mikroszkópos és a későbbi digitalizált képes értékelhetőség érdekében általában festékekkel színezik. A patológiában rutinszerűen először a hematoxinil-eozin (HE) festés készül. Eme festés hatására javul a sejtmagok, a bazofil struktúrák (kékes, lilás-fekete), valamint az általános szövetek és a citoplazmák (halványpiros-rózsaszín) azonosítása. Ezután a mikroszkópos vizsgálat következik. Hosszas évek során alakul ki a szakembereknél olyan tapasztalat, amellyel nagy biztonsággal képesek eldönteni egy metszeti mintáról, hogy az egészséges-e vagy sem. Egy ilyen minta vizsgálata során igen sok szempontot kell figyelembe venni: a nyálkahártya felszínének simaságát, a mirigyek sűrűségét, méretét, párhuzamos irányultságát.



1. ábra: A patológus munkája és szerepe [PATO12]

Nem mindegy azonban, hogy milyen kisegítő lehetőség állnak a szakorvos rendelkezésére. A mikroszkópos vizsgálat során végig kényelmetlen pozícióban ül, a jegyzeteléshez le kell vennie a szemét a mikroszkópról, majd utána újra meg kell találni, hogy hol hagyta abba a vizsgálatot, hisz ennek pontját sem tudta bejelölni. Ezen felül a mikroszkóp folyamatos fókusz beállítása is csak nehezíti a feladatot. Jól látható tehát, hogy segíteni kell a diagnosztizálást végző orvosok feladatát, ha lépést akarunk tartani a növekvő megbetegedések számával. Ennek eredményeképpen született meg a digitális szövettan.

„A digitális szövettani laboratórium olyan technológia, amely lehetővé teszi, hogy a korábban csak a mikroszkópon, illetve az üveg-tárgylemezeken elvégzett szövettani diagnosztikát olyan környezetbe és minőségbe emeljék, amelynek előnye a gyorsabb, pontosabb diagnózis, a mennyiségi paraméterek bevezetése és az interneten keresztüli távdiagnózis” [DELA08].

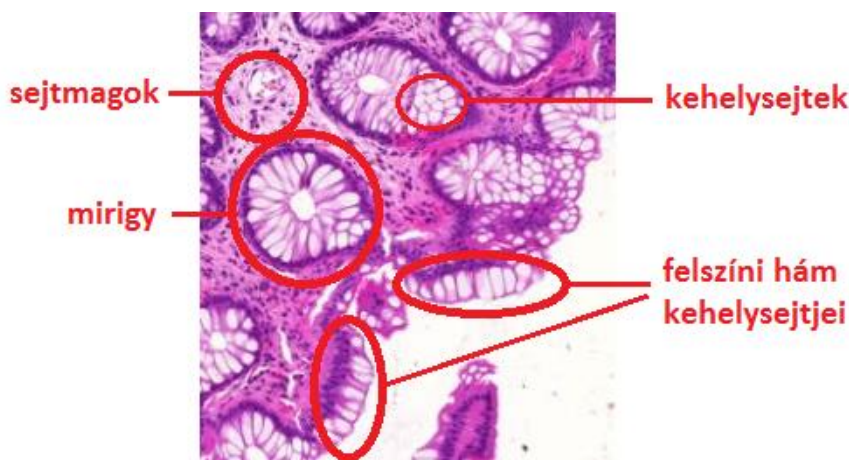
A virtuális mikroszkópiában a virtuális tárgylemez az üveg tárgylemezen lévő minta digitális képe. A digitális kép felbontásának tökéletesnek kell lennie, ugyanazt a minőséget kell látnunk, mint a mikroszkópba nézve. Ez úgy érhető el, hogy egy nagy felbontású kép készítése helyett, akár több tízezer kisebb kép készül a mikroszkóp objektívén keresztül [PATO12].

Tehát a megfelelő képnézegetővel a számítógép előtt ülve, kényelmesebben is felállítható a szövettani diagnózis. A kép bármely pontjához megjegyzések fűzhetők és a fókuszálás problémájával sem kell foglalkozni, hiszen a digitális kép jelen esetben mindig tökéletesen éles. Hogy miért nem terjedt el eddig jobban a digitális szövetanalízis? Ennek több oka is lehet, az egyik talán az, hogy nem mindenhol áll rendelkezésre korszerű számítógép, mely nagy tárhellyel

rendelkezik, ugyanis ezen nagyfelbontású képek mérete a több gigabájtot is elérheti. A terjedésben problémát jelenthet az is, hogy rengeteg olyan orvos van, aki évtizedek óta a bevált módszert követi és kevésbé mutat hajlandóságot az új technológia megismerése iránt.

1.3. A vastagbél felépítése

Ahhoz, hogy jól működő programot tudjunk készíteni, meg kell ismernünk a vastagbél felépítését. A vastagbél több szerkezeti rétegből épül fel, ezek közül a nyálkahártyát kell vizsgálni, hiszen az elváltozások (gyulladás, daganat) innen fejlődnek ki. A 2. ábrán látható szöveti mintán megjelölve láthatóak a feldolgozás szempontjából meghatározó szövettani komponensek. Az ép, és a megbetegedett vastagbél nyálkahártyák közötti különbségek bizonyos szövettani komponensek elváltozásában mutatkoznak meg. Ilyen komponens például a mirigy és az azokon belül elhelyezkedő kehelysejtek, a felszíni hámrétegen elhelyezkedő kehelysejtek és a sejtmagok [KU97].



2. ábra: Szöveti minta felépítése

2. Hasonló rendszerek bemutatása

2.1. Colon

2.1.1. A projekt bemutatása

A projekt az egykori Budapesti Műszaki Főiskola - Neumann János Informatikai karán, azon belül az Informatikai és Automatizálási Rendszerek szakirány keretein belül kezdődött 2008-ban [REM11].

„A problémára megoldást nyújt egy szövettani immundiagnosztikai kiértékelő és speciálisan a vastagbél daganatok differenciáldiagnosztikáját, prognosztikáját támogató szoftver. A program feladata a HE festést kapott digitalizált minták számítógépes feldolgozása a 3DHistech Kft. által több év óta fejlesztett MIRAX szoftver termékcsaláddal együttműködve.

A feldolgozás során a szoftver a morfológiai paraméterek alapján azonosítja az egyes szövetkomponenseket: felszíni hámot, mirigyeket, sejtmagokat. A szöveti képek feldolgozását követően diagnosztikai elemzés készül a szegmentált szövettípusokra a diagnosztikai paraméterek alapján, célzottan az ép, gyulladásos, hyperplasztikus és daganatos elváltozások azonosítására” [REM11].

A készítőik kihasználták a HE festés egy speciális tulajdonságát, mégpedig azt, hogy a sejtmagok a minta festése során jobban kiemelődnek (sötétebbek lesznek) mint a környezetük. Az általuk kínált megoldás egy binarizálási eljárás, amely során egy olyan maszkot készítettek, ami lefedi a sejtmagok teljes területét. Ehhez a szokásos RGB szintér helyett HSV modellbe konvertálták a képet. Azért használták ezt a modellt, mert ez sokkal precízebben írja le a színcapcsolatokat, mint az RGB. Ennek alkalmazásával könnyebben meghatározhatóak azok a paraméterek, amelyekkel maszkolhatják a sejtmagokat.

Mielőtt azonban bármilyen algoritmust is alkalmaznának, szűrniük kellett a megadott képeket, ugyanis a képek zajosak. Sokféle zaj létezik, például a jpg tömörítésből származóak vagy az egyes mikroszkópok fénytörési tulajdonságaiból adódóak. Egyes zajok sajnos nem pontszerű természetűek, hanem a teljes mintára kihatnak, ezért nehezítik a binarizálást. Tehát első lépésként egy zajszűrést alkalmaztak, ekkor azonban fennáll az információvesztés lehetősége, de ebben az esetben ez lényegtelen, mert nem éles határokra volt szükségük.

2.1.2. Sejtmagok detektálása

Szűrés után elkészítettek egy maszkot, amivel szemben követelményeket határoztak meg. A maszk mérete az egész kép méretével megegyezik, ahol sejtmag van ott fehér színű, egyéb más helyeken pedig fekete. A maszk alkalmazása után a sejtmag egyes paramétereit vizsgálták, a sejt területét, kerületét és a morfológiai jellemzőit. A sejtmag területének meghatározásához kitöltő algoritmust használtak. Az algoritmus elve az, hogy megvizsgálunk egy pontot, és ha az egy előre meghatározott szinttartományba esik, akkor átszínezzük, valamint minden, még át nem festett szomszédjára végrehajtjuk ezeket a lépéseket. Iteratív megoldást alkalmaztak, két tárolót definiáltak. Az elsőben (in) a vizsgálatra váró elemek vannak, a másodikban (out) a már megvizsgált elemek halmaza. A tárolókkal lehetett műveleteket is végezni: elem hozzáadás, elem eltávolítás, keresés. Az algoritmus: kezdetben az in tároló csak a kiindulási pontokat tartalmazza az out üreseket. Első lépésben vesznek az in-ből egy elemet és megvizsgálja, hogy annak a pontnak a színe az adott szinttartományba esik-e. Ha igen, akkor a szomszédos pontok közül azokat, amelyek sem az in sem az out-ban sem szerepelnek, beteszük az in halmazba. Utolsó lépésben a megvizsgált elemet minden esetben az outba helyezik. Az outban végzik a paraméterméréseket. A területmérésnél megszámlálják az áthelyezett pixeleket, a pixeleket befoglaló téglalaphoz eltárolják a legkisebb és a legnagyobb x és y koordinátákat és meghatározzák a súlypontot. Az algoritmusuk addig fut, amíg az in halmaz üressé nem válik. Csak a négyes szomszédságot tekintik megfelelőnek, hiszen ez is bejárja a megfelelő területet. A sejtek kerületének meghatározásához egy körvonal detektáló algoritmust használtak. Azt a módszert választották, hogy körbejárják a maszkon és megszámlálják a sejtet határoló pixeleket. A sejtek átmérőit is meghatározták. Mivel a sejtmag általában ovális vagy kör alakú, ezért a sejtmag átlói (a legkisebb és a legnagyobb) jó eséllyel egybeesnek a sejtet befoglaló téglalap átlóival. Első lépésben a sejtmag legnagyobb és legkisebb (x és y) koordinátái alapján meghatároztak egy téglalapot, ami magában foglalhatja a sejtmagot. Meghatározták minden átlónál a ki és belépési pontokat. Az így kapott átlónkénti távolságokat már csak egyszerű minimum és maximumkereséssel kell vizsgálni.

2.1.3. Mirigyek detektálása

Mirigyek detektálásánál azt használták ki, hogy minden mirigyet sűrű sejtmagsor határol, valamint felhasználták a fentebb ismertetett sejtmagtérképet. A mirigyek pozícióját és elhelyezkedését a sejtmagok egymáshoz viszonyított helyzetéből próbálják meghatározni. A teljes mintát ábrázoló képből indultak ki, mégpedig olyan módon, hogy a már korábban meghatározott sejtmagok helyére pár pixeles foltot rajzolnak, így olyan maszkot kaptak, ahol az üresen hagyott foltok helyén mirigyek lehetnek. Hogy tényleg mirigy helyezkedik el a megfelelő helyen, azt úgy döntenek el, hogy megméri a méretét. Ez azonban sajnos nem ad olyan pontos eredményt, amivel további számításokat lehetne végezni. További pontosítás úgy történt, hogy egy hálót [FI09] készítettek annak érdekében, hogy a megfelelő maszkot megkapják és utána a

szükséges méréseket végrehajthassák. A háló készítéséhez felhasználták a korábbi sejtmagtérképet. A sejtmagok közepéből csigavonalas bejárást indítanak adott távolságig. Az algoritmus addig megy, amíg nem talál egy másik sejtmagot, vagy egy fekete pixelt. Ha talál, akkor egy vonalat húznak a két pont közé. Ezeket a lépéseket minden lehetséges sejtmagra végrehajtják. Csigavonalas bejárást azért alkalmaznak, mert így a legközelebbi sejtmagot találják meg legelőször. A nem pontosan meghatározott közepétől elindítják a kitöltő algoritmusukat, így jó közelítéssel egy helyes maszkot kapnak. A végén egy dilatációt hajtanak végre.

2.1.4. Hámdetektálás

A hám detektálásánál is kihasználták a sajátos tulajdonságát, hogy a hámot alkotó sejtmag csak a minta szélén helyezkedhet el. Ezért legelőször a minta körvonalát határozták meg. A mirigydetektálásnál használt algoritmust alkalmazták, vagyis elkészítik a sejtmaghálót, majd az egész háló által körbehatárolt területet kitöltik. A körvonal detektáló algoritmust átalakítva alkalmazzák. A módosítás annyiból áll, hogy el tudják tárolni azt, hogy a kontúrpont a hám része vagy sem. Első próbálkozásra úgy gondolkodtak, hogy a hám azon tulajdonságát használják ki, hogy a sejtmagok sűrűn helyezkednek el. Megpróbálták kizárni az alacsony sűrűségű helyeket. Ilyen helyek tipikusan a metszés során megsérült helyeken keletkeznek, ez azonban nem elég pontos a detektáláshoz, hiszen előfordulhatnak néha a szakított részen is magas sejtmag sűrűségű területek. Tehát egy pontosabb eljárást alkalmaztak, kihasználva azt, hogy a hámot alkotó sejtek hasáb alakja miatt, a sejtmagokat egyenletes plazma veszi körbe. Az algoritmus a következő: első lépésben binarizálják a mintát úgy, hogy az elkészült maszk a minta teljes egészét lefedi. Ezt összehasonlítják a kitöltött hálós képpel, ahol a kettő határa egybeesik, ott nem lehet hám, mert hiányzik a plazmaréteg [REM11].

2.1.5. A projekt értékelése

A Colon projektnél a kép méretére való tekintettel tették meg az első optimalizációt. A felhasználó számára lehetővé tették a vizsgálandó mintarész kijelölését, valamint az algoritmusokat úgy írták meg, hogy megvizsgálják a bemenő adat méreteit és amennyiben az túlságosan nagy, akkor fel kell bontani kisebb egységekre. Ezeket az egységeket akár később, külön szálon lehet feldolgozni, vagy esetleg valamilyen más párhuzamos megoldást alkalmazni, ami ilyen nagyméretű képeknél jelentősen gyorsíthatja a feldolgozási időt. A projekt a párhuzamosítható feladatokat GPU segítségével oldotta meg.

A projekt előnye, hogy rámutat bizonyos hibákra a tesztelési fázisban. Az RGB, vagy HSV színkezelés közötti különbség jól látható. A feldolgozó algoritmusuk RGB használatával 36%-kal több sejtmagot detektált, mint amennyi a képen van.

A Colon projekt számos problémára felhívta a figyelmünket, mint például a sejtmagdetektálás hibái. A tesztelés fázisát követően bebizonyosodott, hogy az algoritmusok, amiket alkalmaztak a

sejtmagok detektálása során, több találatot produkáltak, mint amennyi a valós sejtmagszám. Az implementált algoritmusainak a futási ideje alkalmanként órákban mérhető. Mindent összevetve a projekt valamilyen hatékonysággal detektálja a sejtmagokat, bizonyos algoritmusokat érdemes lehet kipróbálni.

2.2. Automatizált rákdiagnosztika hisztopatológiai képek alapján

2.2.1. Előfeldolgozás

A [DEYE12] rendszer első fázisa a zajcsökkentés. Egy egyszerű technikát alkalmaznak, mégpedig a küszöbölést, amit háttérkorrekció és szűrés követ. Küszöbölésre egy olyan Otsu algoritmust alkalmaznak, mely automatikusan megtalálja a legoptimálisabb küszöbértéket. A háttérkorrekció csökkenti a különböző mintákon tapasztalható, eltérő fényviszonyokból adódó eltéréseket, a szűrés pedig segít a zajcsökkentésben. A zajcsökkentés után a sejtek detektálása következik, melynek két megközelítését is alkalmazzák.

2.2.1.1. Régió alapú sejtdetektálás

A régió alapú sejtdetektálásban ismét küszöbölést alkalmaznak. A küszöbölés ugyan elhatárolja a sejteket a háttértől, de nem tudja elszeparálni az egymással összefolytakat. Morfológia és a watershed [DELA08] algoritmus megoldás jelenthet erre a jelenségre. A watershed algoritmus képes szétválasztani az „összenőtt” sejteket, de fontos megjegyezni, hogy problémát jelenthet a túlszegmentálás. A túlszegmentálás ellen aluláteresztő szűrőt alkalmaznak, és csak a megjelölt mirigyekben engedélyezik a szeparálást.

A küszöbölés után törlik a kis objektumokat és a nagy területeket, majd befoltozzák a megmaradt lyukakat, javítva ezzel a szegmentálás eredményét. Ehhez morfológiai operátorokat is alkalmaznak négyzet, vagy nyolcszög alakú struktúráló elemmel. Fontos az elem méretének helyes megválasztása, ugyanis kisebbnek kell lennie, mint a sejt mérete, így alkalmazásakor a nála kisebb objektumok azonnal eltávolíthatóak.

A másik a régió alapú megközelítés, egy tanuló algoritmus, ami eldönti, hogy a pixel egy sejthez tartozik vagy sem. Az algoritmus a pixelekből kinyert jellemzők alapján tud szelektálni. Ilyen jellemzők lehetnek a szín vagy textúra információk. Például k-közép algoritmussal két csoportba oszthatóak a pixelek, sejt és nem sejt csoportba. A hátránya ennek a módszernek, hogy emberi beavatkozás szükséges, ugyanis manuálisan történik az osztályozás.

2.2.1.2. Határkeresésen alapuló sejtdetektálás

Az első próbálkozásban manuális szegmentálást alkalmaztak, ami annyit jelent, hogy a felhasználó megjelöl néhány pontot a sejtmag határán, majd ezeket a pontokat felhasználva, egy zárt alakzat rajzolható ki. A módszer problémája, hogy igen sok munkát igényel, főleg ha nagy mennyiségű sejtről van szó.

Az automatikus határdetektáláshoz aktív kontúr modelleket, „snakes” (kígyókat) alkalmaztak. A kígyók alatt itt deformálható geometriai hálókat értünk. A cél, hogy ezek a hálók felvegyék egy sejt alakját, mely alak pontossága függ a kontúrponktoktól, de néha még így is szükség lehet a felhasználó beavatkozására [DEYE12].

A határ és a régió alapú megközelítésnek is van előnye és hátránya is egyaránt. A határvonal megközelítés precízebben határozza meg a sejtek alakjait a régió alapúval szemben, viszont meg kell határozni a határpontokat, mely folyamat manuális beállítást igényelhet.

2.2.2. Jellemzők vizsgálata

A diagnózis felállításához a szöveti minta sejt struktúrájában lévő eltéréseket vizsgálják, melyek mérésére morfológiai, texturális, fraktális és/vagy az intenzitás alapú jellemzőket használnak fel. Ezen jellemzők a sejtek pontos helyének meghatározása után nyerhetők ki.

2.2.2.1. Morfológiai jellemzők

A morfológiai jellemzők információt nyújtanak a sejt méretéről és alakjáról. A méret kifejezhető a sugárból, a területből és a sejt kerületéből. A forma meghatározza a tömörséget, simaságot, a fő- és mellékátlót, a homorúságot, a kerületet, valamint a szimmetriát. Ezen mérőszámok alapján megfelelő hatékonysággal eldönthető egy sejtről, hogy egészséges-e vagy sem.

2.2.2.2. Texturális jellemzők

A textúra olyan összefüggő pixelhalmaz, amely folyamatosan ismétlődik egy adott képen. Segítségükkel leírhatóak a felület intenzitásában megjelenő változások. Ezen jellemzők leírására két, széleskörben alkalmazott mátrixot használtak fel:

- Együttes előfordulási mátrix
Ez a mátrix számszerűsíti a különböző texturális jellemzőket, mint például a korrelációt, kontrasztot, a szürkeárnyaltos képpontok közötti térbeli függőségek felhasználásával.
- Run-length mátrix
Ez a mátrix a textúra durvaságát méri, meghatározott irányokban.

2.2.2.3. Fraktál-alapú jellemzők

A fraktál egy olyan objektum, amely önhasonló tulajdonsággal rendelkezik, azaz különböző nagyítás mellett is változatlan alakúnak látszik. A fraktál geometria információt ad egy objektum gyakoriságáról, komplexitásáról, számszerűsítve annak önhasonlósági szintjét. Ezen információkhoz ki kell számolni az objektum fraktáldimenzióját, ugyanis ez a leggyakrabban használt jellemző.

A fraktál analízis alkalmazása segíthet megérteni a különböző jelenségeket a rákdiagnosztikában.

2.2.2.4. Topológiai jellemzők

A topológiai jellemzők információt szolgáltatnak a szövet struktúrájáról és a sejtek térbeli eloszlásáról. A Voronoi diagramok és azok Delaunary háromszögelései használhatók az egymással szomszédos sejtek közötti függőségek felderítésére. A szöveti képen a Voronoi diagram egy konvex poligont képez minden egyes sejtre, amivel meghatározhatnak számos paramétert, mint például a minimális feszítőfát, úgy, mint a teljes fa hosszát [DEYE12].

Kiszámolják az élek valószínűségét az éleket összekötő csúcsok euklideszi távolságához mérten. Ezt felhasználják a Waxman modellben [WAX88]. A Waxman modell paramétere határozza meg a kapcsolatokat a sejtgráfban. Ezzel a lokális és globális gráf tulajdonságai is meghatározhatók.

A gráfban a jellemzők a következők: rang, csomópont együttható és az excentricitás (a legnagyobb távolság az adott csúcs és bármely más csúcs között). Ezeket a paramétereket a globális gráfon is le lehet mérni, majd ezekből következtetéseket vonni.

2.2.2.5. Intenzitás-alapú jellemzők

Meghatározásuk a kép szürkeárnyaltos és színhisztogramjából lehetséges. Ez a fajta jellemző nem szolgáltat információt a pixelek térbeli eloszlásáról. Weyn definiált egy optikai sűrűség jellemzőt [WEWO98], amit a szürkeárnyaltos kép intenzitás értékekből határozott meg. Átkonvertálták a sejtmag minden pixelét egy megfelelő értékre, a következő képlet alapján: $OD = \log_{10}(\text{háttér intenzitás érték} / \text{pixel intenzitás érték})$. Ezekután kiszámolták az optikai sűrűség összegét és vették a középértéket, ami a sejtmagot egy intenzitás jellemzővel írja le. Ez a művelet elvégezhető az RGB és a YIQ színterek esetén is.

2.2.2.6. Jellemző kiválasztás

Nagyszámú jellemző határozható meg, azonban ezeknek csak a töredékét lehet felhasználni az osztályozási folyamat során, ezért ezek számát célszerű valahogy csökkenteni.

A jellemzők között kétféle keresést lehet végrehajtani. Első lehetőség, az egyes jellemző alcsoportok vizsgálata. Ez a megközelítés jó lehet abból a szempontból, hogy ha létrejön az eredmény, akkor minden lehetséges jellemzőt felhasználva, a legjobb összeállítás érhető el, viszont rengeteg jellemzőt kell felhasználni a számítások során és ez nagymértékben ront a feldolgozási időn.

A másik keresés a szekvenciális keresés. Ezzel a módszerrel folyamatosan törölhető, illetve hozzáadható új jellemző az objektumhoz, egészen addig, amíg jobb kombinációt már nem lehet létrehozni.

2.2.3. Diagnózis

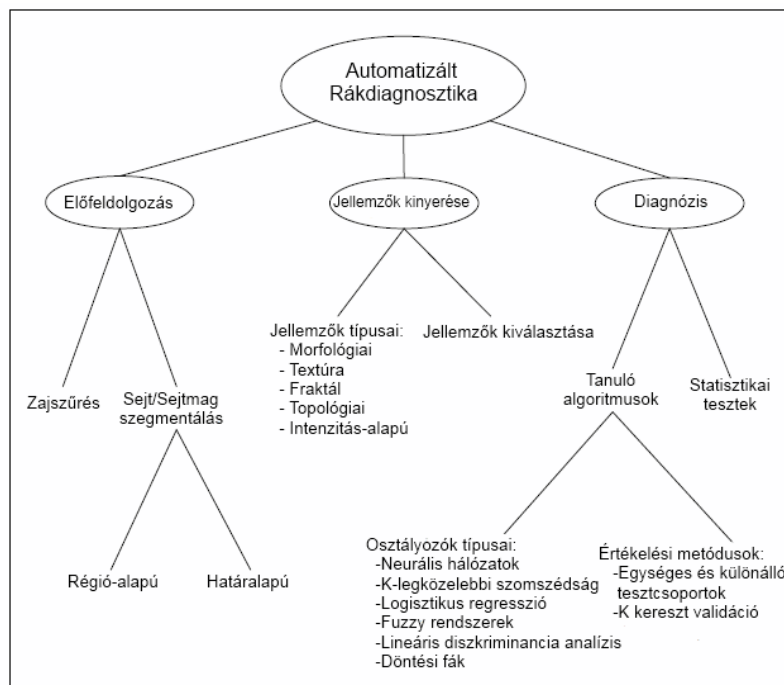
A megfelelő jellemző kiválasztása után, amellyel dolgozni fog az algoritmus, a következő lépés, rosszindulatú struktúrák elkülönítése a többtől. E lépésnél a sejt vagy szövet osztályozva lett, mint rákos, kezdeti beteg, vagy egészséges. A tanulmányok egy csoportja a meghatározott jellemzőket statisztikailag vizsgálta. A statisztikai teszteknel azonban független minták tételezhetők fel, és e szerint vonhatók le a megfelelő következtetések. Az adatok erősen függenek ugyanazon személy, különböző szöveti képtől, ami ebben az esetben nem teszi lehetővé a független mintákkal való statisztikai összehasonlítását, mert ez esetben félrevezető lehet az eredmény.

A tanulmányok egy másik csoportja különböző tanuló algoritmusokat használ fel a diagnózis felállításához, majd ezekkel az eszközökkel próbálják az osztályozást elvégezni [DEYE12].

Néhány ezek közül az algoritmusok közül:

- Neurális hálózat
- K-legközelebbi szomszédság
- Fuzzy rendszerek
- Döntési fák

A rendszer működésének vázlata a 3. ábrán látható.



3. ábra: A rendszer működésének vázlata [DEYE12]

2.2.4. A projekt értékelése

Az előfeldolgozási fázisban sok rendszerhez hasonlóan küszöbölés és zajszűrés megy végbe. A sejtek szegmentálására watershedet alkalmaznak, aminek a hátránya a túlcsordulás, és ezt folyamatosan korrigálni kell.

A diagnózis felállításához igen sokfajta jellemzőt vizsgálnak, aminek ugyan a feldolgozási ideje nagyon hosszú lehet, de cserébe többféle betegség felismerésére is alkalmassá válhat a program. További hátrány még, hogy az egyes lépések manuális beállítást igényelnek, amik a feldolgozás idejét lassíthatják.

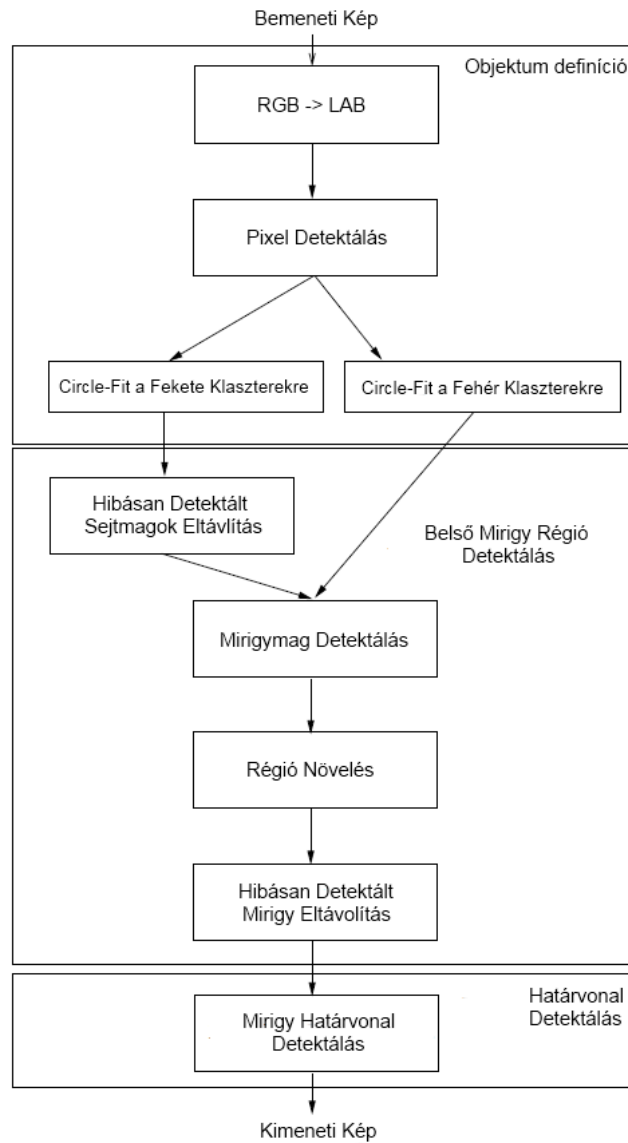
2.3. Vastagbél mirigyek szegmentálása objektum gráf használatával

2.3.1. Áttekintés

A feldolgozandó kép egyszerű objektumokra való bontásával, majd azok térbeli elhelyezkedéseit figyelembe vételével meghatározhatóak a mirigyek pontos helyei [KA08].

A pixeleket először különböző szövettani struktúrákba kell csoportosítani, amely tulajdonság a színintenzitásukból következtethető. A képpontokat további körkörös, egyszerű objektumokká kell szétbontani a Circle-Fit algoritmus segítségével (a fehér objektumot a luminális régiók, a fekete objektumok pedig a sejtmag régiók), és így felépíthető az a gráf, amelyből számszerűsíthető a térbeli elhelyezkedésük. A gráfból megtudhatóak olyan jellemzők, amelyek segítségével meghatározhatóak a mirigyeken belül található sejtmagok. Ezt követően egy másik ábrát is felépítünk a sötétebb objektumok összekötésével, majd a belső mirigy régiókat is meg lehet határozni régiónöveléssel, amely a mirigymagokból indul és a gráf határáig tart.

A hibásan detektált mirigyek eltávolítása, a belső régiókból vett jellemzők segítségével történik. Végül a valós mirigyek körbehatárolása következik, amelyek a megfelelő felhámsejtek és a belső mirigyrégiók egyesítésével megy végbe. A folyamat blokkdiagramja a 4. ábrán látható.



4. ábra: A rendszer működésének blokkvázlata [KA08]

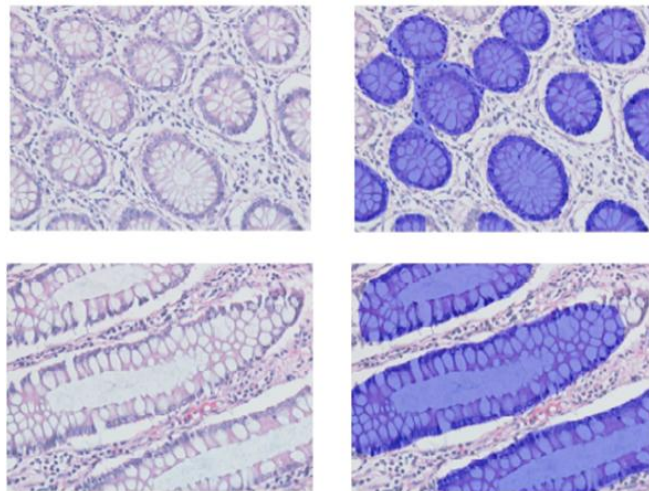
A projekt részletes bemutatása a Függelékben található.

2.3.2. A projekt értékelése

A projekt, ezelőtt még nem alkalmazott megközelítéseket mutat be, amelyek miatt egyedülállónak nevezhető. Volt már olyan tanulmány, amely használt például körillesztő algoritmusokat bizonyos régió szegmentációhoz, de ez az első eset, amikor a körillesztés a program elsődleges és alapvető eljárása és a segítségével meghatározott egyszerű objektumok a későbbiekben részletesen elemezendő komponensek.

A LAB színtér használatával könnyebbé vált az osztályozás és az elkülönítés. Ezen okból kifolyólag a saját megvalósításunk során is e színtérben végezzük el a különböző szövettani struktúrák elemzését.

Mivel a funkciók nagyrészt nem implementáltuk a saját rendszerünkben, emiatt csak az irodalomban említett eredmények alapján lehet a teljesítményt értékelni. Az 5. ábrán látható két teszteredmény, bal oldalt az eredeti, jobb oldalt a program által detektált mirigyek. Míg a felsőnél szabályos kör alakúak, addig az alsón hosszúkas elnyúlt mirigyek vannak. Figyelemre méltó, hogy az algoritmus helyes működését nem befolyásolja a detektálandó komponensek alakjai, ellenben szabályos, egymástól viszonylag távol levő mirigyek esetén is adhat helytelen eredményt (5. ábra, jobbfelső kép).



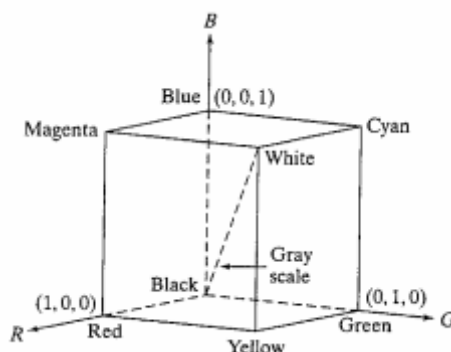
5. ábra: Teszteredmények [KA08]

3. Használandó módszerek, algoritmusok, keretrendszerek

3.1. Színterek

RGB:

Az egyik leggyakrabban alkalmazott színtér. A színek felírhatók három alapszín keverékeként. Ez a három alapszín a vörös (R), zöld (G) és a kék (B). Ezt a színteret úgy is elképzelhetjük, mint egy háromdimenziós kockát. E képzeletbeli tér tengelyei az egyes színkomponenseknek felelnek meg (6. ábra).

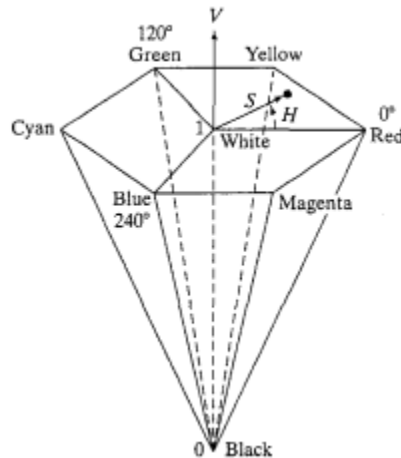


6. ábra: RGB sematikus ábra [GOWO03]

HSV:

- Hue (színárnyalat): a szín típusát adja meg, amely lehet piros, kék vagy sárga
- Saturation (telítettség): a szín telítettsége
- Value (érték): a szín értéke

A HSV színtér egyik kedvelt megjelenítési módszere a gúla (7. ábra), amelyben a színek egy háromdimenziós gúlában „helyezkednek” el. Az árnyalat a gúla körkörös keresztmetszete, a telítettség a középponttól való távolsága, és az érték pedig a csúcsos végétől való távolsága [SCCO87].



7. ábra: A HSV gúla [GOW003]

Feladatunk során különböző színű és árnyalatú mintákkal kell dolgoznunk. Ha RGB színteret használnánk, rendkívül nagy változtatással járna a különböző színek felismerése. Ennek kiküszöbölésére HSV színtérben tároljuk képeinket. Így pusztán egy koordináta megváltoztatásával világosabb vagy hasonló fényerő mellett, de más telítettségű képre is működhetnek a detektáló algoritmusaink.

LAB:

- A: a piros-zöld skála leírója
- B: a kék-sárga skála leírója
- Luminancia (fényesség): A harmadik, független dimenzió

Az 'A' és 'B' koordináták együttesen határozzák meg magát a színt, függetlenül annak világosságától. E színtérben a nagyon hasonló árnyalatú pixelek is jól elkülöníthetőek [MAC67].

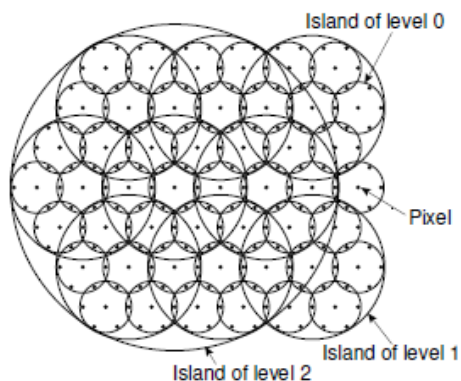
3.2 Szín Struktúra Kód

A feldolgozandó digitális szöveti kép igen sok homogén területet és zavart tartalmaz, amelyek hátráltatják a mirigyek sikeres detektálását, szegmentálását. Az előkészítő fázisban egy olyan algoritmust alkalmazunk, amely a HSV színtérben egymástól kevésbé eltérő árnyalatokat összemosza úgy, hogy a lényegesen eltérő színek önálló objektumokká szeparálódjanak. Ennek hatására a mirigyek fehéres, világos belseje jól elkülöníthető lesz.

A színek szegmentálására a Szín Struktúra Kód (Color Structure Code, továbbiakban CSC) algoritmust implementáltuk.

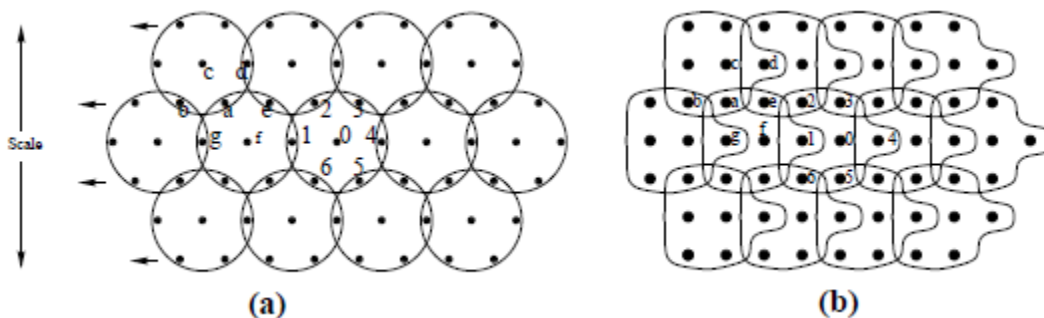
3.2.1 Az algoritmus felépítése

A CSC alapját a Hartmann féle hierarchikus régiónövelő módszer adja, amelyben a 8. ábrán is látható, hierarchikus topológiát [HA87] az úgynevezett „szigetek” alkotják, különböző szinteken. A legelső szinten lévő sziget 7 pixelből áll (egy kiválasztott középső pixel és a körülötte lévő 6 pixel). A következő szinten már nem 7 pixelből épül fel ez a hexagonális elem, hanem az előző szinteket felépítő szigetekből. Tehát, ha az első szinttől különböző szinten vagyunk, akkor egy sziget, 7 darab, a hierarchiában egyel lejjebb lévő szigetből épül fel. Ennek a struktúrának az építése egészen addig folytatódik, amíg egyetlen sziget magába nem tudja foglalni az egész képet [PRST12].



8. ábra: Hexagonális sziget hierarchia [PRST12]

A gyakorlatban ez a hexagonális struktúra azonban nem alkalmazható, ugyanis a fényképezőgépek nem ilyen formában rögzítik a pixeleket, hanem ortogonális alakban. Ahhoz, hogy megkapjuk ezt a formát, a hatszög első és harmadik sorát el kell tolni balra. Az eltolás és annak eredménye a 9. ábrán látható.

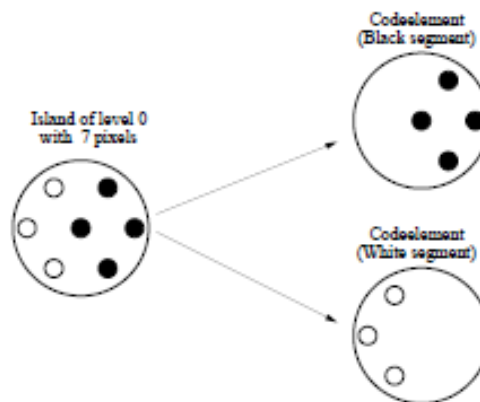


9. ábra: A hexagonális (a) és az ortogonális (b) struktúra [PRST12]

Az algoritmus működése

3.2.2.1 Inicializálás

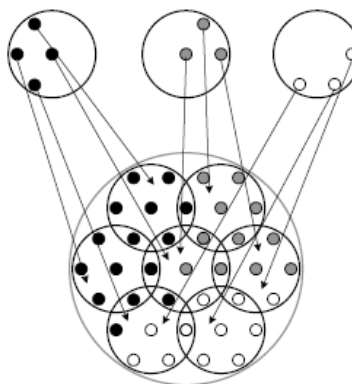
Ebben a fázisban az algoritmus, az első szintet alkotó szigeteket bontja fel homogén régiókra a 10. ábrán látható módon. A folyamat során kezdeti kódelemeket kell létrehozni, amelyek akkor jönnek létre, ha egy szigeten belül találhatóak olyan homogén pixelek, amelyek egymás mellett helyezkednek el. Homogén alatt itt azt értjük, hogy az egymással összehasonlított pixelek színkülönbsége nem halad meg egy előre megadott küszöbértéket. A bejegyzésekben eltárolódik még információ az összefüggő terület alakjáról, és színeinek középértékéről.



10. ábra: Egy nulladik szintű sziget felbontása két homogén régióra [PRST12]

3.2.2.2 Összekapcsolás

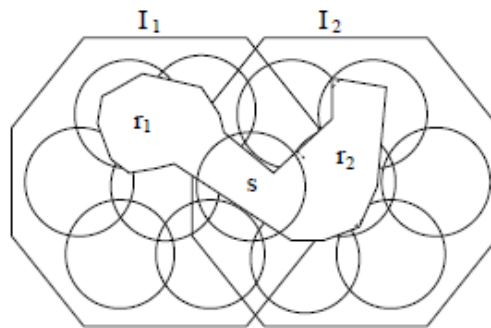
Az összekapcsolás fázisban az n . szinten lévő szigetek kódelemeiből építjük fel az $n+1$. szinten lévő új kódelemeket. Akkor jön létre új kódelem az $n+1$. szinten, ha vannak olyan egymás mellett lévő szigetek az n . szinten, melyek metszéspontjaiban homogén színek kapcsolódnak egymáshoz (lásd 11. ábra).



11. ábra A kódelemek összekapcsolása [PRST12]

3.2.2.3 Szétvágás

A legtöbb régiónövelő algoritmusnál fennáll az a probléma, hogy a lassan változó színeket is egy szegmensbe sorolják. Ezért dolgoz fel a CSC algoritmus lokális és globális adatokat egyaránt a szegmentáláshoz. A gyakorlatban ez itt úgy valósul meg, hogy az összekötött kódelemek között további ellenőrzések mennek végbe minden egyes szinten. Ha a színeltérés nagyobb a szintek között, mint az előre megadott küszöbérték, akkor a továbbiakban megszakad a kapcsolat a két bejegyzés között. A 12. ábrán látható, hogy bár r_1 és r_2 elméletileg egy régióba tartozik, a túl nagy színeltérés miatt el kell szeparálni őket egymástól.



12. ábra r_1 és r_2 régió, közös s alrégióval [PRST12]

3.3 Soros kapcsolódó komponens módszer

A soros kapcsolódó komponens módszer egy a bináris képek szegmentálására használt algoritmus [FIPE03]. Megkeresi egy kép összefüggő komponenseit és mindegyikhez egy egyedi színazonosítót rendel. A végeredmény tehát egy színes kép, amelyen az azonos színű képpontok jelentik az összefüggő komponenseket. Első lépés, hogy a képpontokon értelmezünk egy szomszédsági relációt. Két képpont szomszédja egymásnak, ha legfeljebb egy pixel távolságra helyezkednek el egymástól. Vizsgálhatunk négyes és nyolcas szomszédságokat. Soros megvalósítás esetén, az algoritmus szekvenciálisan bejárja a képet, és minden lényeges képpontot ('1'-t tartalmazó pixelt) megjelöl egy inkrementális jelöléssel az alábbi szabályok szerint (négyes szomszédság esetén):

$$R_4(p) = L \text{ ha } \{ ((x-1,y) = 0, (x,y-1) = 0) \text{ vagy } ((x-1,y) = L, (x,y-1) = 0) \text{ vagy } ((x-1,y) = 0, (x,y-1)) = L \}$$

$$R_4(p) = L \text{ ha } \{ (x-1,y) = M, (x,y-1) = L \} \rightarrow (L=M);$$

ahol, p egy képpont, x és y jelenti a p pixel x, y koordinátáját.

Tehát meghatározzuk az egy osztályba tartozó jelöléseket és ezen ekvivalenciaosztályok minden elemeinek közös címkét adunk.

3.4. Aforge.Net

Az Aforge.net egy ingyenes matematikai és képezelő algoritmus gyűjtemény, amelyben bizonyos, általunk használni kívánt függvények is megtalálhatóak. Főként mesterséges intelligencia és képfeldolgozáshoz szorosan kapcsolódó eljárásokat tartalmaz. E gyűjtemények között találhatók különböző sebesség optimalizált szűrők, HSV, HSL, RGB alapú színterek, forgatások, átméretezések, alakzatok megtalálására, megszámlálására, kiterjedésük mérésére használható függvények, kiválóan működő neurális hálózat, továbbá fuzzy és genetikai algoritmusok.

3.5. OpenCV

Képfeldolgozó körökben az egyik legnépszerűbb algoritmus gyűjtemény az OpenCV. Alapvetően C++ nyelvben íródott, de az EmguCV wrapper segítségével .NET környezetben is lehetséges a használata. Az előre implementált éldetektáló és jellemzőkereső algoritmusai miatt van szükségünk a használatára.

4. Saját rendszer bemutatása

4.1. Célkitűzés

Hosszútávú tervünk egy diagnosztikai célú szoftver előállítás, amely a vastagbél daganatok gyorsított felismerését segíti. Különböző színintenzitású, árnyalatú és kontraszt nagyságú hematoxilin és eozin festésű szöveti minták elemzésére alkalmas program, amely gyorsan kiértékeli a felhasználó által beadott mintát. Mindazon algoritmusok implementálása, amelyek együttesen meghatározzák egy mirigy helyzetét, jellemzőit, majd el is tárolják ezeket az információkat. Továbbá szükséges még az egyéb szövetkomponenseket (sejtmag, felszíni hámréteg) felismerő és osztályozó algoritmusok implementálása, az egyes modulok paraméterezését automatizáló algoritmusok megvalósítása és a rendszerhez tartozó grafikus felhasználói felület elkészítése

4.2. Rendszervázlat

A rendszervázlat átfogó összesítést ad arról, hogy milyen folyamatok során valósítottuk meg a mirigyek és a sejtek szegmentálását, valamint ezek feldolgozását. Ismerteti a feldolgozás általános folyamatát és a rendszer lépéseit a legmagasabb absztrakciós szinten. A rendszer működésének vázlata a 13. ábrán látható.



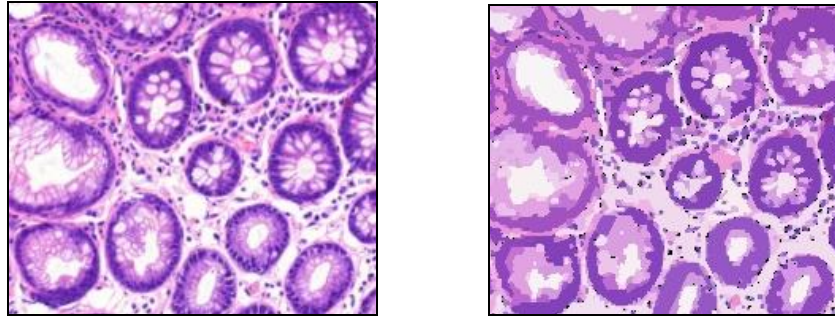
13. ábra: A rendszervázlat blokkvázlata

4.3 A rendszer működésének bemutatása

4.3.1 Mirigyetektáláshoz megvalósított funkciók

4.3.1.1 Szín Struktúra Kód (Color Structure Code)

A Szín Struktúra Kód felhasználásával az azonos színű pixeleket összemossuk, így az esetek többségében jól elkülönülnek a mirigyek határvonalai a nyálkahártya egyéb részeitől (14. ábra). Az algoritmus részletes bemutatása a 3.2 pontban olvasható.



14. ábra: Bal oldalon az eredeti kép, jobb oldalon a színszegmentálás által transzformált kép

4.3.1.2 RGB-ből LAB színtérbe konvertálás

Azáltal, hogy a feldolgozandó képet a szín alapú szegmentálás után LAB színtérbe konvertáljuk, egy olyan képet kapunk, amelyen főként homogén háttérterületek és a mirigyek körvonalai találhatóak. Nagymértékben megkönnyítik a későbbi feldolgozást, hiszen független a szöveti mintán használt kontrasztanyag típusától és erősségétől (15. ábra).

4.3.1.3 Küszöbölés

Ha a homogén részek átlag pixeleinek és az ettől eltérő pixelek átlagainak arányából meghatározunk egy küszöbértéket, az eredmény egy, csak mirigykörvonalakat tartalmazó kép. Közel 50 szöveti képet teszteltünk a megfelelő küszöbérték meghatározásának érdekében. Egyértelműen megállapítható, hogy a LAB színtérben elegendő csak az 'A' paraméter szerint küszöbölni, mivel e paraméter arányaiban mindig ugyanakkora értékkel tér el a mirigy belső és a mirigy határ régiói között. A kapott eredményt binarizáljuk abból a célból, hogy fekete-fehér képpontokat kapjunk, amely fontos a későbbi feldolgozás szempontjából (15. ábra).

4.3.1.4 Összekötés

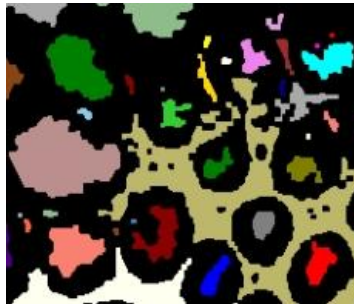
Bizonyos körvonalak tartalmazhatnak a homogén tartományhoz közeli színintenzitású pixeleket, ezért a küszöbölés következtében a körvonalak nem alkotnak zárt alakzatot (ideális esetben kört). Erőziót használunk az összefüggő komponensek létrehozására, majd dilatációt a vastagság helyreállítására (15. ábra).



15. ábra: A küszöbölés előtti LAB szintérben levő kép (bal oldal), a küszöbölés utáni kép (középső) és az összekötött komponensek (jobb oldal)

4.3.1.5 Összefüggő komponens szegmentáció

A megkapott mirigykörvonalak behatárolják a képen található mirigyek nagyrészét. A hátralévő feladat az elkülönítés és azonosítás. Soros kapcsolódó komponens módszer minden összefüggő ponthalmazhoz egyedi színeket rendel, amelyek segítségével azonosíthatók a későbbiekben (16. ábra). Minden egyes, így módon megtalált komponensre területszámítást végzünk, majd a túl nagyokat (háttér) és a túl kicsiket (egyéb, nem mirigyhez tartozó részeket) figyelmen kívül hagyjuk.



16. ábra: Az összefüggő komponens módszer alkalmazása után

A széleken levő mirigyeket nem sikerült detektálni. Ennek oka, hogy ezeknek nincs minden irányból határvonala, ebből kifolyólag nem zárt alakzatok, így nem detektálhatóak, mint összefüggő komponensek. A probléma megoldásának érdekében szükséges tehát egy újabb lépés bevezetése. A legkézenfekvőbb, ha minden, a képen található szöveti mintázatnak létrehozunk egy zárt határvonalat.

4.3.1.6 Határvonal létrehozás és illesztés

A határvonalat detektáló algoritmus pseudokódja:

Eljárás HatárvonalDetektálás(Kép, N)

```
Szélesség <- Kép szélesség
Méret <- Kép méret
Param <- LegkisebbKerület(Kép)
ÚjKép <- Kép nagyságú új kép
Kép <- ÁtlagSzűrő(Kép)
Kép <- GaussSzűrő(Kép)

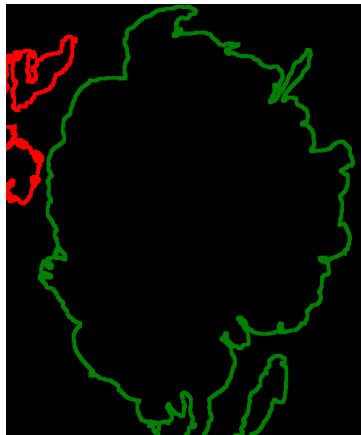
Ciklus i <- 1-től N-ig
    Ha Kép[i]<245 akkor
        Kép[i]=255
    Különben
        Kép[i]=0
    Elágazás vége
Ciklus vége

Ciklus i<- 1-től N-ig
    Ha Kép[i] Kép szélén elhelyezkedő pixel és Kép[i]=255 akkor
        ÚjKép[i ]=255
    Egyébként
        Ha (Kép[i]=255 és (Kép[i+Szélesség]=0 vagy Kép[i-Szélesség]=0 vagy
        Kép[i+1]=0 vagy Kép[i-1]=0))
            ÚjKép[i]=255
        Különben
            ÚjKép[i]=0
        Elágazás vége
    Elágazás vége
Ciklus vége

ÚjKép <- ÖsszefüggőKomponensAnalízis(ÚjKép, Param)
ÚjKép <- Dilatáció(ÚjKép,2)
HatárvonalDetektálás <- ÚjKép
```

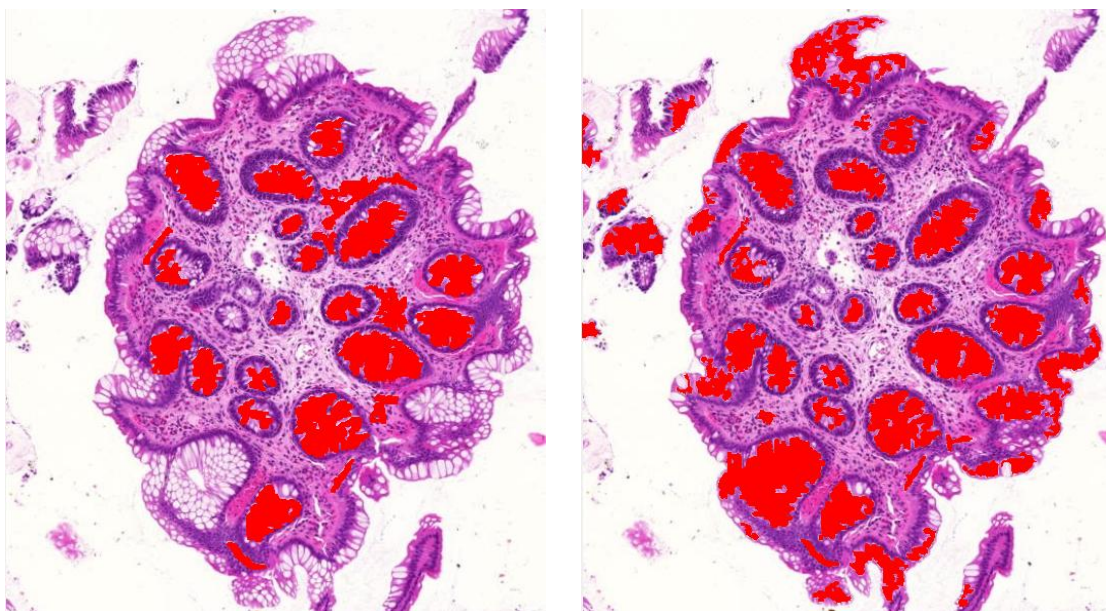
Eljárás vége

Első lépésben átlagolósűrő és Gauss-sűrő alkalmazásával kiküszöböljük a képen levő zajokat. Az első ciklus feladata minden háttér képpontot feketére és minden nem háttér képpontot fehérre állítani. A küszöbértéket 30-40 szöveti mintán elvégzett kísérlet során állapítottuk meg és végül 4%-os hiba határértéket (245) választottunk, így a halványabb részek sem tűnnek el. A második ciklus lényegében egy olyan éldetektáló, amely négyes szomszédság alapján határoz meg egy adott képpontot, lekezelve azon eseteket, amikor a kép szélén található érdemleges képpont, azaz az elemezendő terület bizonyos része kilóg a képből. Ezek után összefüggő komponens analízist alkalmazunk. A módszer második paramétere jelenti azt a minimális nagyságú területet, amelyen kívül eső képpont csoportokat figyelmen kívül hagy a függvény. E paraméter értékét a Param változó jelenti. A Param egy olyan érték, amely meghatározható a képen található szöveti minták közül a legkisebbik területének a nagyságával. Megkaptuk a minták körvonalát, amelyen utolsó lépésként kettes erősségű dilataciót hajtunk végre, hogy megfelelő vastagságúak legyenek a meghatározott határvonal görbék, ennek eredménye a 17. ábrán látható.



17. ábra: A szöveti mintázatok határvonalai

Az eredményt ráillesztjük arra a képre, amelyet a 4.3.1.4-es lépés (összekötés) után kaptunk és erre, az újonnan létrejött képre végezzük el az összefüggő komponens vizsgálatot. A végeredmény képen (18. ábra) látható, hogy szinte minden, a kép szélén elhelyezkedő mirigyjelölt a hozzájuk tartozó felszíni hámréteg kehelysejtjeivel is detektálva lett.

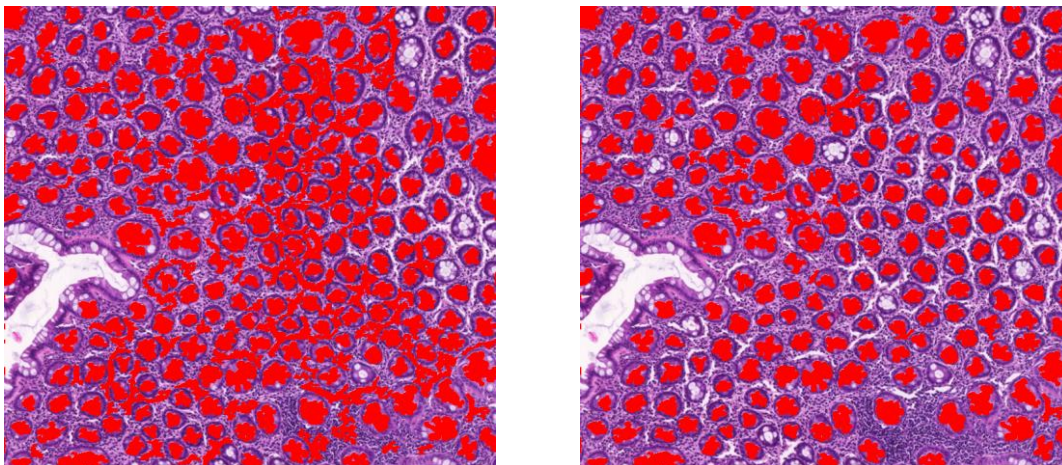


18. ábra: Bal oldalon az eredetleg detektált mirigyek, jobb oldalon a kiszámolt határvonal létrehozással detektált mirigyjelöltek

4.3.1.7 Hibásan detektált mirigyek eltávolítása

A színszegmentálás segítségével igen nagymértékben lehet megtalálni a valódi mirigyeket, de sajnos a módszer velejárója, hogy arányaiban véve sok, nem mirigy is detektálva lett. Ezen hibásan detektált mirigyek kiküszöböléséhez, a már detektált régiókat és a sejtmagos régiókat hasonlítjuk össze, ugyanis az algoritmus arra a feltételezésre épül, hogy a mirigyen belül nem található jelentős mennyiségű sejtmag.

Az algoritmus a következőképpen működik: adottak az összefüggő komponensek, amelyek a mirigyjelölteket tartalmazzák. Ezután a sejtmag régiókat tartalmazó képet felépíti az algoritmus. A két képen képpontról képpontra halad a feldolgozás, és az összehasonlítás során a program megszámlolja, hogy egy adott komponens, hány darab fekete pixelt, azaz sejtmag területnyi képpontot tartalmaz. A feldolgozás végén még kiszámítjuk, hogy egy komponens, a területéhez viszonyítva, milyen arányban tartalmaz sejtmagot. Ha ez az arányszám egy előre megadott, manuálisan beállított küszöbértéknél nagyobb, akkor a komponens nem mirigyrégió, tehát eldobjuk. A feldolgozás eredménye a 19. ábrán látható.



19. ábra: Bal oldalon az alap mirigydetektálás eredménye a javító algoritmus nélkül, jobb oldalon pedig a hibásan detektált mirigyek eltávolítására készült algoritmus segítségével kapott eredmény

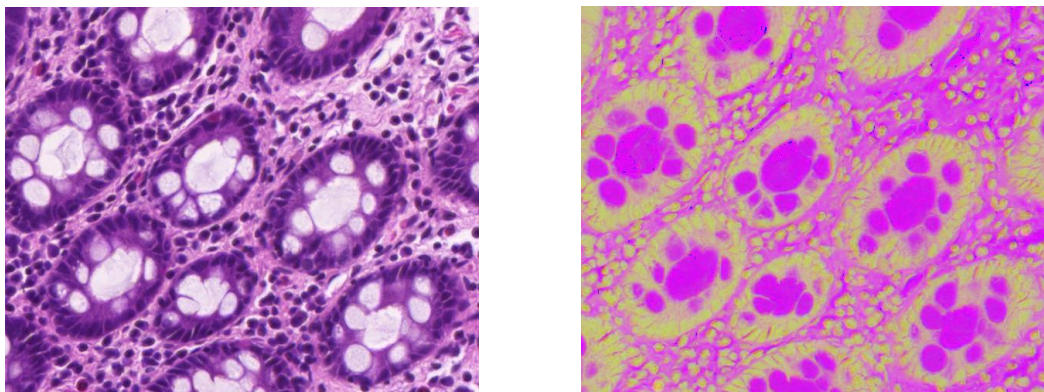
A módszer hatásfoka nagyban függ a sejtmagdetektálás hatékonyságától. E tanulmány két sejtmagdetektálási módszerről tesz említést, a HSV és a színszegmentálás alapú módszerekről. A gyakorlatban a színszegmentálás alapú alkalmazható nagyobb hatékonysággal a hibás mirigyek eltávolításánál, viszont ennek az ára az, hogy a 4.3.2.2 pontban is kifejtett, mirigyen belüli, rosszul detektált sejtmagok megzavarhatják a feldolgozást, és a jól detektált mirigyek is elvetődnek.

4.3.1 A sejtmagdetektáláshoz megvalósított funkciók

A sejtmagok elhelyezkedésének, formájának, adott területre jutó darabszámának meghatározásával számos következtetést lehet levonni, amelyek mind a detektálást, mind a diagnózist elősegítik. A megvalósított sejtmagdetektálási algoritmusok jelen pillanatban igen kezdetlegesek, egyelőre csak a helytelenül detektált mirigyek eltávolításához használhatóak.

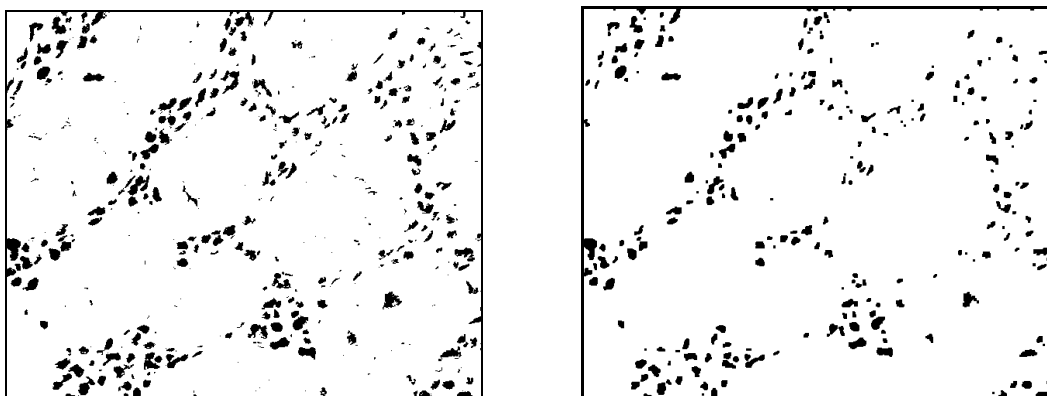
4.3.1.1 Sejtmagdetektálás HSV színtér használatával

Első megközelítésben a sejtmagok nagyságát és jellegzetes sötét színét figyelembe véve történt a detektálás. Az RGB színtérben történő színalapú számítások igen nehezen kivitelezhetőek, ezért a képet HSV színtérbe kell konvertálni, amelynek eredménye a 20. ábrán látható.



20. ábra: A baloldalon az eredeti RGB, a jobb oldalon a HSV-be konvertált kép

HSV színtérben a sejtmagok kiemelkedően magas szaturációs értékkel rendelkeznek, ezért egy jól megválasztott küszöbölési intervallummal kinyerhetőek. Ennek érdekében egy olyan adaptív algoritmust implementáltunk, amely a binarizálás alsó küszöbét folyamatosan változtatva, kicsi, sejtmag méretű kapcsolódó komponenseket keres. Az algoritmus azt a küszöbintervallumot fogja kiválasztani, amely a legtöbb komponenst adja eredményül. Az így kapott képen, lásd 21. ábra, zajszűrést végzünk mediánszűrő segítségével.

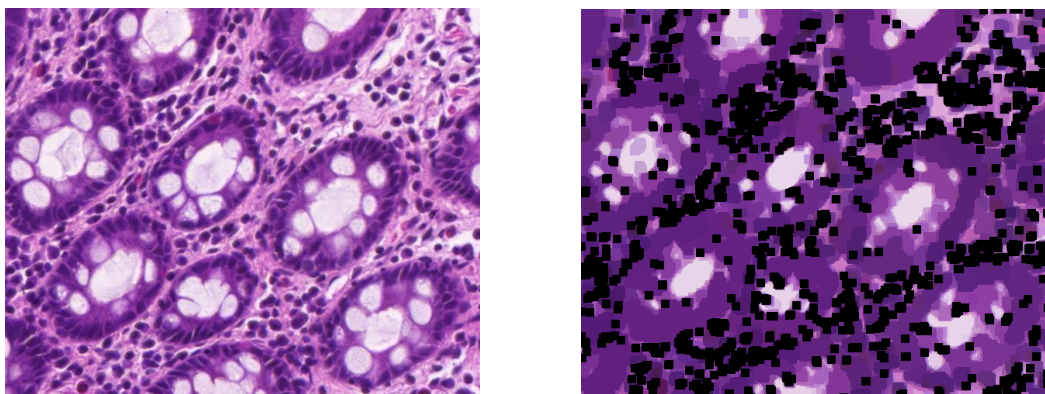


21. ábra: Bal oldalon a binarizált kép zajszűrés előtt, jobb oldalon zajszűrés után

Ennek a megoldásnak előnye az, hogy nagyon kevés a hibásan detektált sejtmag, a hátránya pedig, hogy az egymáshoz túlságosan közel elhelyezkedő sejtmagok nagyobb méretű, összefüggő komponenst alkotnak, tehát a méret alapján történő szűrés során elvesznek.

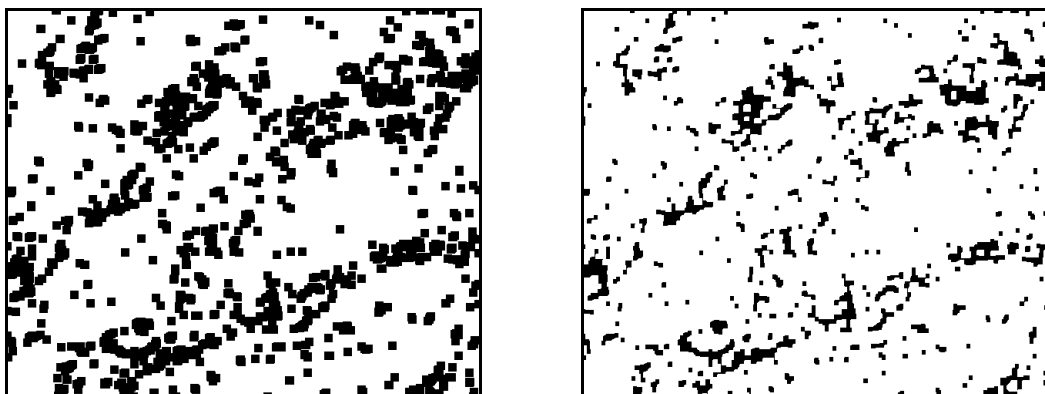
4.3.1.2 Sejtmagdetektálás színszegmentálás alkalmazásával

Egy másik megközelítésben a sejtmagdetektáláshoz a 4.3.1.1 fejezetben említett Szín Struktúra Kódot alkalmaztuk. Az algoritmus lefuttatása után jól látható, hogy a sötét sejtmagok körül fekete színszegmensek jelennek meg. Ezen szegmensek kiemelhetőek erózió használatával, amelynek végeredménye a 22. ábrán tekinthető meg. Az így kapott képpontok csak a sejtmagok pozíciójának leírására szolgálnak, a formájukról és nagyságukról nem adnak információt.



22. ábra: Bal oldalon az eredeti kép, jobb oldalon a színszegmentált, erodált kép

A színszegmentált, erodált képből binarizálással kinyerhetők a fekete képpontok, majd zajszűrést alkalmazva eltüntethető a nem sejtmagokhoz tartozó rész nagy százaléka, lásd 23. ábra.



23. ábra: Bal oldalon a binarizált kép zajszűrés előtt, jobb oldalon zajszűrés után

Ennek a megközelítésnek az előnye az, hogy a sejtmagos régiók nagyobb pontossággal határozhatóak meg, hátránya pedig az, hogy a mirigyeken belül lévő kehelysejtek körül is megjelenhetnek a fekete képpontok, és ezen elemek nehezen távolíthatóak el teljes mértékben, zavarva ezzel a későbbi elemzéseket.

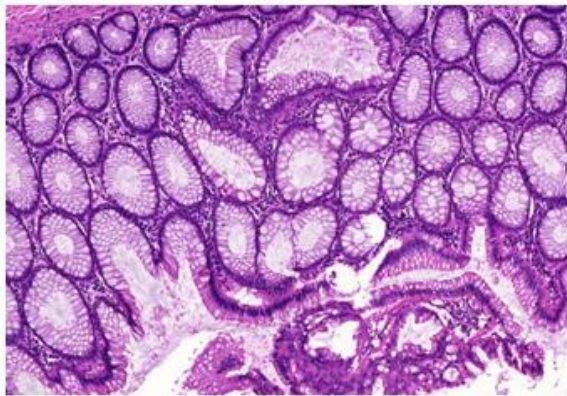
4.3 Tesztelés és eredmények bemutatása

4.3.1 Mirigydetektálás eredményei

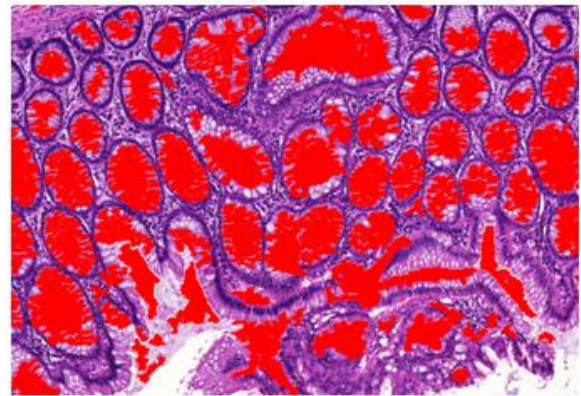
A program fő funkciója a mirigyek detektálása. A tesztelés során a detektálás hatékonyságát mutatjuk be. Dolgozatunkban öt reprezentatív képen végezzük el az elemzést, amelyek mindegyike rendelkezik valamilyen egyedi, működési hatékonyságot befolyásoló jellemzővel.

A 24. ábra (a) és (c) képen a mirigyek nagyon közel helyezkednek el egymáshoz, és igen különböző méretűek. A 24. ábra (e) és a 25. ábra (a) képen, a szöveti minta szélén számos

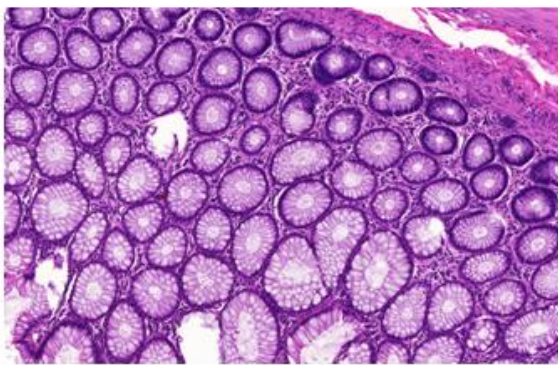
elmetszett mirigy található, emellett 24. ábra (e) kép belsejében bizonyos helyeken hiányoznak vagy igen halványak a sejtmaggal teli régiók, amely nehezíti a helytelenül detektált mirigy eltávolító folyamat sikeres végrehajtását. A 25. ábra (c) képen balról jobbra egyre világosabb színűvé válik minden szöveti komponens.



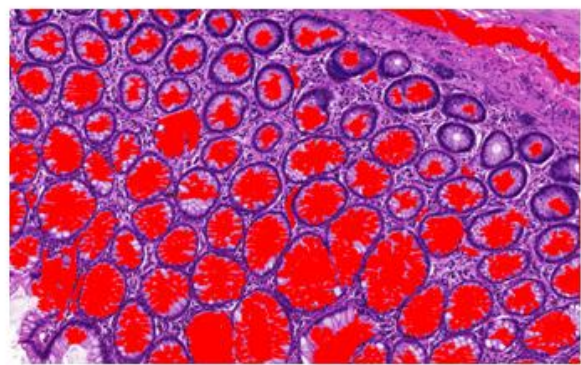
(a)



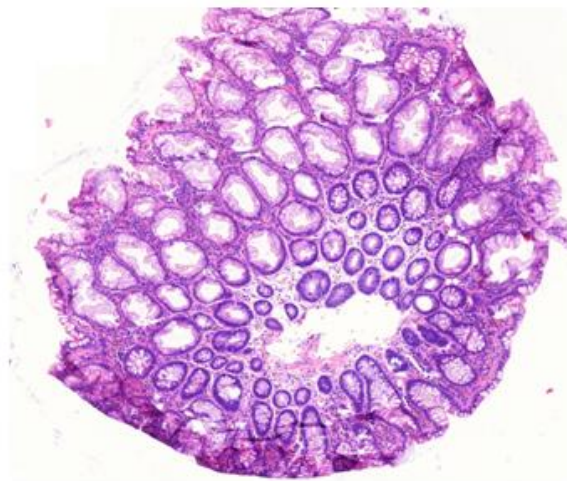
(b)



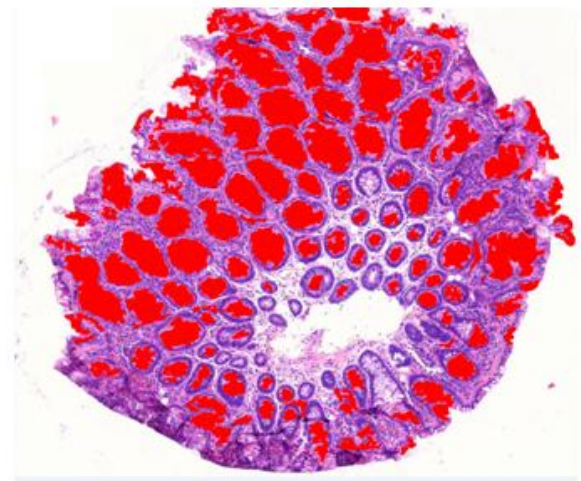
(c)



(d)



(e)

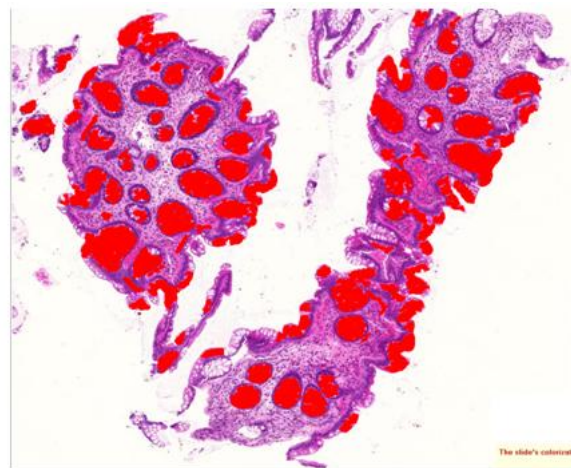


(f)

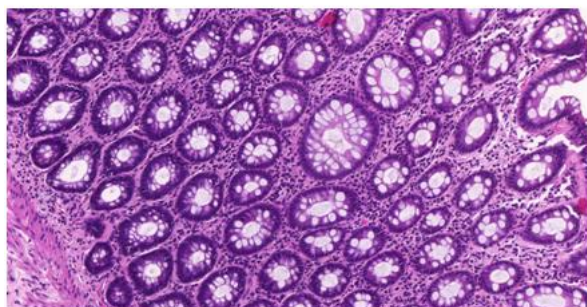
24. ábra: (a, c, e): Az eredeti szöveti minták, (b, d, f): az analízis utáni kép



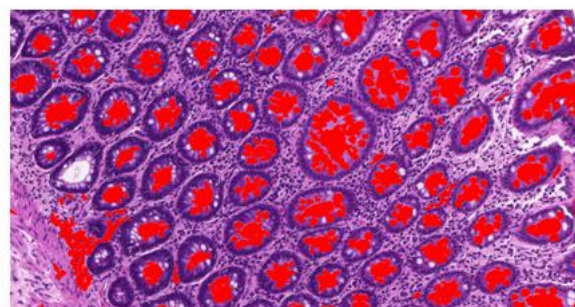
(a)



(b)



(c)



(d)

25. ábra: (a, c): Az eredeti szöveti minták, (b, d): az analízis utáni kép

Ahogy az 24. és 25. ábrán levő eredmény részképeken látható, minden esetben közel az összes mirigyet sikerült detektálni, ezzel együtt viszont néhány olyan régiót is detektáltunk, amelyekről biztosan tudjuk, hogy nem mirigyekhez tartoznak.

A felismerés hatékonysága nagyban romlik azon területeken, ahol alacsony sűrűségű sejtmag rész található, mert hibásan, mirigyként detektálódnak, tehát a korrigáló algoritmus eredménye nem megfelelő ezen régiókon, lásd 25. ábra (d) bal alsó rész. További hatékonyságot befolyásoló tényező, ha a kehelysejtek határai viszonylag sötét színűek, ekkor ugyanis külön komponensként különülnek el. Bizonyos esetekben képhibák is megzavarhatják a helyes működést, elsősorban a szöveti mintából kiszakadt részek, amelyek mirigyként detektálunk, a 24. ábra (c) részképen láthatunk ilyen eseteket. A tesztelés eredményei összefoglalva a 26-28. ábrán láthatóak.

A rendszer hatékonyságának vizsgálatához a következő mérőszámok kerültek bevezetésre [EI05] [LU04]:

- **Hamis pozitív (HP):** a detektált mirigyek halmazában található, valójában nem mirigyek, és a detektált mirigyek összegének az aránya, százalékban kifejezve
- **Igaz pozitív (IP):** a detektált mirigyek halmazában található valódi mirigyek, és a detektált mirigyek összegének az aránya, százalékban kifejezve
- **Hamis negatív (HN):** a detektált mirigyek halmazában nem található valódi mirigyek, és a detektált mirigyek összegének az aránya, százalékban kifejezve
- **Igaz negatív (IN):** projektünk esetében ez a szám a HN értékből következtethető, kiszámítása a következőképp történik: $(100 - HN)$
- **Precízió:** a helyesen detektált mirigyek és az összes talált mirigy számának aránya.

Kiszámítása:
$$\frac{IP}{IP+HP}$$

- **Érzékenység:** a program azon képességét határozza meg, hogy az milyen hatásfokkal képes megmondani a pozitív eredményeket, azaz esetünkben azt, hogy a detektált mirigy, milyen valószínűséggel lesz valódi mirigy.

Kiszámítása:
$$\frac{IP}{IP+HN}$$

- **Sajátosság:** a program azon képességét határozza meg, hogy az milyen hatásfokkal képes megmondani a negatív eredményeket, azaz esetünkben azt, hogy milyen valószínűséggel nem detektálja, a ténylegesen nem létező mirigyeket.

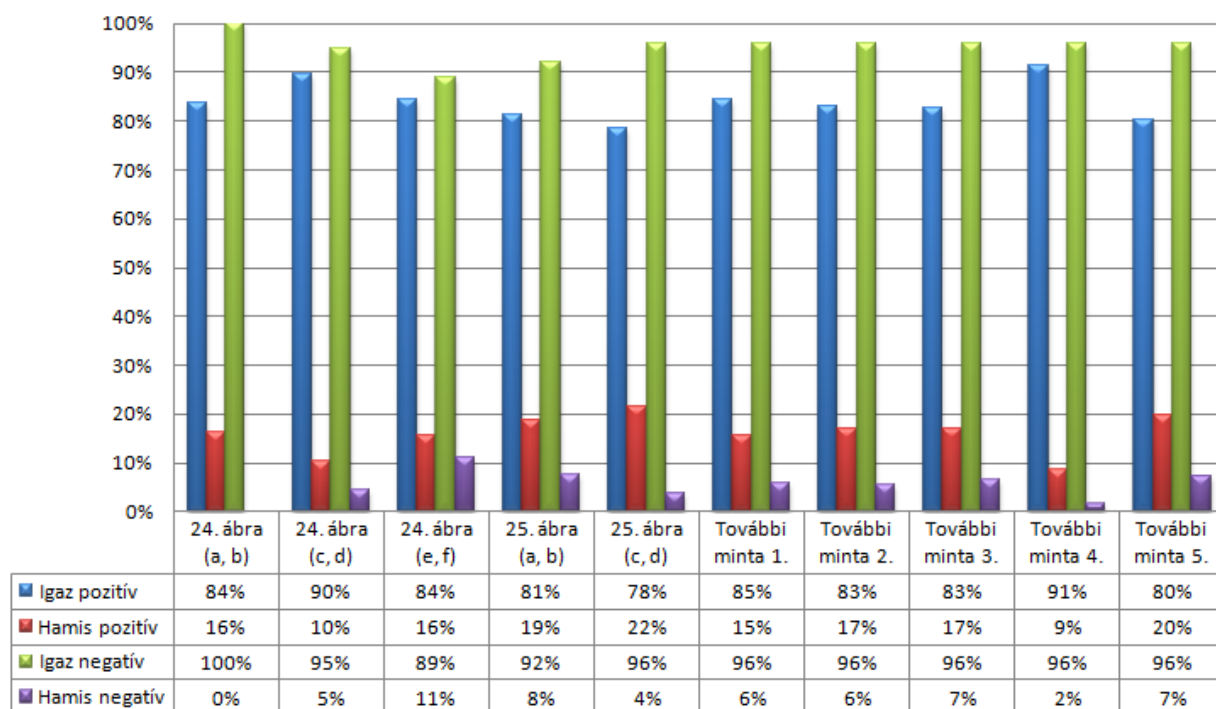
Kiszámítása:
$$\frac{IN}{IN+HP}$$

- **Pontosság:** megadja az IP és IN, és az IP, IN, HP, HN összegének arányát, azaz, hogy milyen pontossággal sikerült detektálni a valódi mirigyeket, a hibás detektálásokkal együtt számítva.

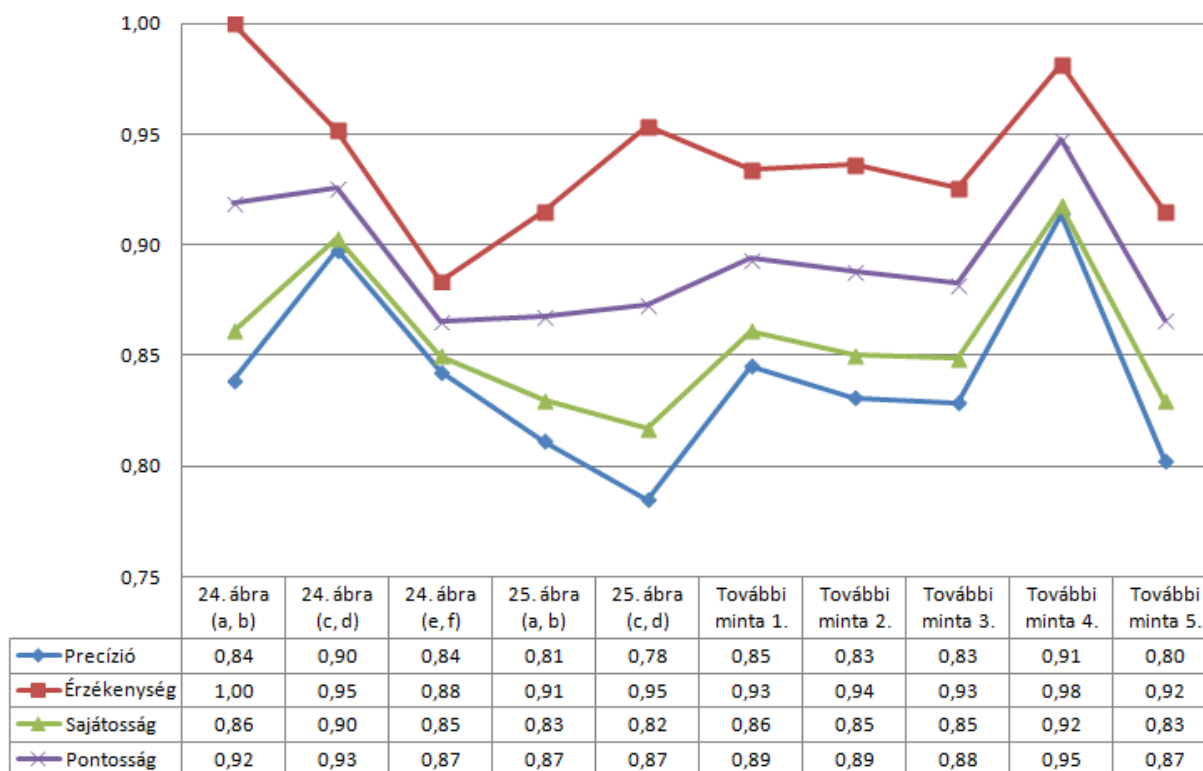
Kiszámítása:
$$\frac{IP + IN}{IP + IN + HP + HN}$$

	Összes valódi mirigy	Összes detektált mirigy	Helyesen detektált mirigyek	Hibásan detektált mirigyek	Nem detektált mirigyek
24. ábra (a, b)	52	62	52	10	0
24. ábra (c, d)	83	88	79	9	4
24. ábra (e, f)	103	108	91	17	12
25. ábra (a, b)	47	53	43	10	4
25. ábra (c, d)	65	79	62	17	3
További minta 1.	76	84	71	13	5
További minta 2.	63	71	59	12	4
További minta 3.	94	105	87	18	7
További minta 4.	54	58	53	5	1
További minta 5.	71	81	65	16	6

26. ábra: A tesztelési eredményeket tartalmazó táblázat



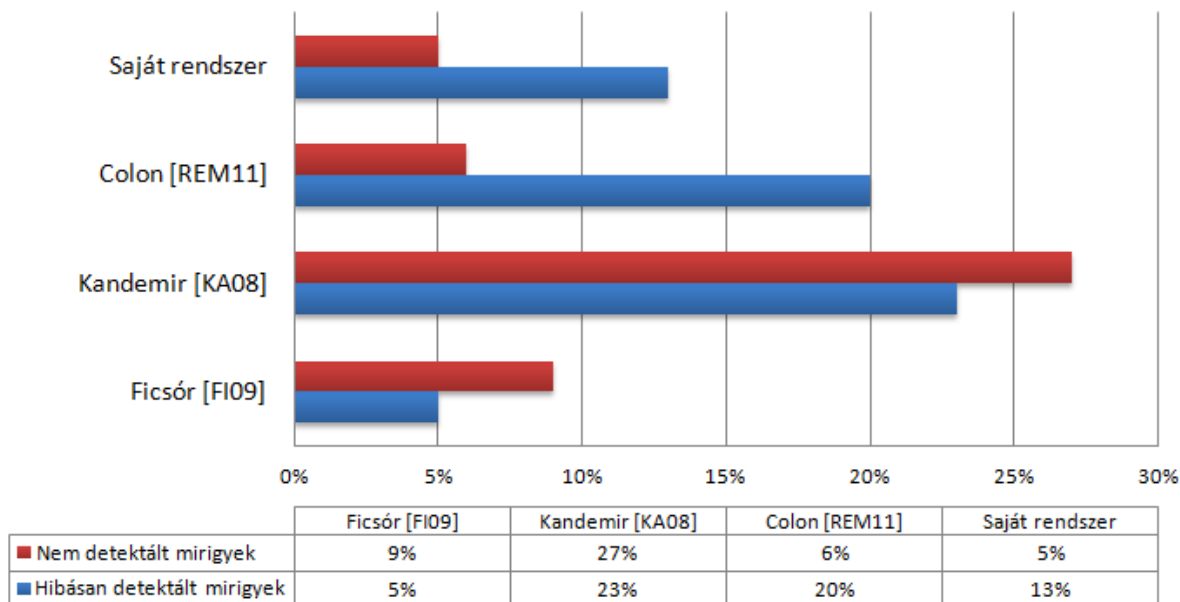
27. ábra: A hibaarány



28. ábra: Precízió, érzékenység, sajátosság, pontosság

A 27. ábráról leolvasható, hogy igen csekély mértékben marad olyan valós mirigy, amelyet nem talál meg a program, ez a szám a HN értékből következtethető. Cserébe viszont a hibája az, hogy igen magas a HP érték is, amely arra utal, hogy relative sok hibásan detektált objektum is található az elemzett képen. A 28. ábrán található mérőszámok közül a pontosság emelendő ki, ugyanis a legfontosabb paraméter a teszt szempontjából. Az értéke átlagosan 0,89, amely azt jelenti, hogy kb. 89%-os pontosságú mirigydetektálás érhető el a program jelenlegi állapotával.

4.3.1 Az eredmények összehasonlítása más rendszerekkel

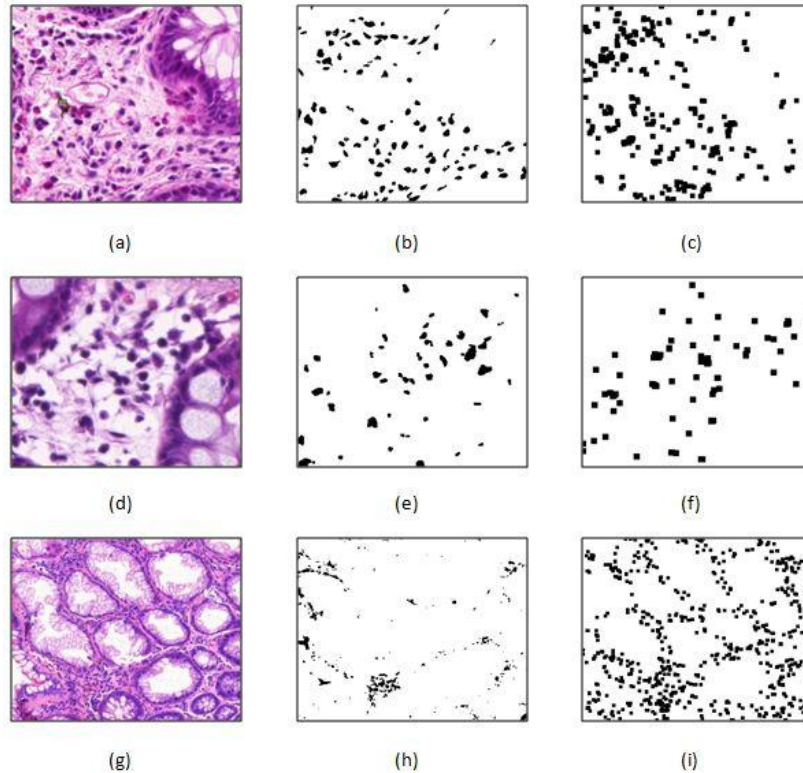


29. ábra: más rendszerek eredményeivel való összehasonlítás

Rendszerünk eredményeit az irodalomkutatás során megismert hasonló rendszerek [REM11, KA08, FI09] értékeivel is összehasonlítottuk (29. ábra). Ezek közül a mi rendszerünk találja meg legnagyobb hatékonysággal a mirigyeket. Ugyan nálunk némileg több a hibásan detektált, mint Ficsór [FI09] publikációjában, de lényegesen kevesebb, mint a további rendszereknél.

4.3.2 Sejtmagok detektálásának tesztelése

A dokumentumban említett mindkét algoritmust (HSV alapú és színszegmentálás alapú) teszteltük. Három olyan képet választottunk, amelyeken jól látható a két módszer előnye és hátránya. A 30. ábrán látható a tesztelés eredménye. Az (a, d) képen jól elkülöníthetők a háttértől a sejtmagok, látható, hogy mindkét algoritmus igen pontos eredményt produkált (b, e és c, f kép). A (g) képen a sejtmagok egymáshoz való távolsága jóval kisebb az előző képekhez viszonyítva. A HSV küszöbölés módszer a nagyon közel lévő sejtmagokat elvetette, mert egybefüggő komponensként érzékelt ezen területeket, és ezek nagysága többszöröse egy sejtmag átlagos méretének. A színszegmentáló módszer jóval pontosabb eredményt adott ebben az esetben.



30. ábra: (a, d, f) Az eredeti szöveti minták, (b, e, h) a HSV küszöbölt sejtmag detektálás eredménye és (c, f, i) a Szín Struktúra Kód segítségével detektált sejtmagok

4.4 Továbbfejlesztési lehetőségek

Ahogy a tesztelésből is kiderült, az elsődleges továbbfejlesztési szempont a hibásan detektált mirigyeket kiszűrő algoritmus továbbfejlesztése. A mostani tézisünk szerint, ez egy magasabb hatásfokú sejtmagdetektálással érhető el, amivel kiküszöbölhetőek azok a hibásan detektált mirigyek, amelyek valójában összeolvadt sejtmagos régiók. Egy másik korrekciós módszer lehet textúra alapú megközelítés alkalmazása. Feltevésünk szerint, betanított minták segítségével felismerhetővé tehetőek a valójában nem mirigyhez tartozó területek.

További feladat egy olyan régiónövelő algoritmus implementálása, amely segítségével teljes egészében kitölthető a helyesen detektált mirigyek területei, és így tényleges alakjukat lehet majd elemezni a pontos diagnózis felállításához.

Implementálandó, az eltérő festési eredmények okozta szöveti minták közötti szignifikáns színelkülönbségek kiküszöbölését szolgáló, automatikus festési séma felismerés és az ezekhez beállított, ideálisan paraméterezett metódusok.

Miután a fent említett funkciók magas megbízhatósággal működnek a szöveti képek túlnyomó többségére, elkészíthető a diagnosztizálás funkció is.

Összegzés, a projekt értékelése

A program jelenlegi működése a következőképp foglalható össze: a betöltött képen egy színszegmentáció hajtódik végre a HSV színtérben a Szín Struktúra Kód algoritmus segítségével, amely végeredményül egy olyan képet kapunk, ahol a mirigyek elkülönülnek a többi szöveti struktúrától. Az összes mirigy egyedenkénti azonosításához további metódusok szükségesek.

A színszegmentált képet LAB színtérbe konvertáljuk, majd azon küszöbölést hajtunk végre, amelyhez a küszöbértéket 50 elemzett kép utáni tapasztalat alapján határoztuk meg. Az eredményül kapott, binarizált képen megfigyelhetőek nyitott mirigyjelölt komponensek. Ezen nyitott elemek bezárására erodálást alkalmazunk, amely nagymértékben megnöveli a zárt komponensek számát. Ahhoz, hogy a szöveti képek szélén található mirigyeket is megtaláljuk, egy általunk fejlesztett algoritmussal meghatározzuk a szöveti képek körvonalait, majd ezen vonalakat ráillesztjük az előző pontban kapott binarizált képre. Az így kapott képen megkeressük az összes összefüggő komponenst, amely komponensek a mirigyjelölteink lesznek. A detektált objektumok közül azonban nem mindegyik elem ténylegesen mirigy. Ezen hibásan megjelölt mirigyek kiválasztásához és eltávolításához szintén kifejlesztettünk egy algoritmust.

A hibásan megjelölt mirigyek eltávolításához először a sejtmagokat tartalmazó régiókat kell meghatároznunk. A sejtmagdetektáláshoz kétfajta módszert is kidolgoztunk. Az első módszerben a szöveti képet HSV színtérbe transzformáljuk, majd küszöböljük, ugyanis a sejtmagok szaturációja kiemelkedik a többi szöveti struktúra szaturációs értékei közül. Az eredményül kapott képen megkeressük azokat az összefüggő komponenseket, amelyek mérete nem nagyobb egy sejtmag várható méreténél. Ez a módszer akkor ad pontos eredményt, ha a sejtmagok élesen elhatárolódnak egymástól, máskülönben összefolynak a binarizáció során, és méretük miatt kiszűrjük ezen területeket az összefüggő komponens keresésnél.

A másik sejtmagdetektáló módszer kiindulása az első lépésben is említett színszegmentált kép. A rajta végzett erodáció után információkat gyűjthetünk a sejtmagok elhelyezkedéseiről, ugyanis a sejtmagok körül fekete képpontok jelennek meg. E módszer előnye, hogy nem szükséges a sejtmagok elszeparáltsága, hátránya viszont, hogy sok fals eredmény is keletkezik a mirigyeken belül a kehelysejtek metszéspontjainál. A tapasztalat azt mutatta, hogy a színszegmentált képen végzett sejtmagdetektálás nagyobb hatékonysággal működik, így ezt a módszert használjuk fel a hibásan detektált mirigyek eltávolításához.

Az általunk fejlesztett hibásan detektált mirigyek eltávolításához tehát már adott az előző bekezdésben tárgyalt sejtmagos régiókat tartalmazó kép. Algoritmusunk arra a feltételezésre épül, hogy a mirigyeken belül kevés sejtmag található. A mirigyjelöltek komponenseit

tartalmazó képet hasonlítjuk össze a sejtmagos régiókat tartalmazó képpel. Minden egyes komponensnél feljegyezzük, hogy az hány sejtmagrégiós pixelt tartalmaz. Az összehasonlítás után kiszámoljuk a komponensek területeire eső sejtmagok számát. Ha ez az arány nagyobb egy manuálisan, előre meghatározott küszöbértéknél, akkor a komponenst eltávolítjuk. A hibásan detektált mirigyek eliminálása után, az algoritmus berajzolja a szűrés után megmaradt komponenseket az eredeti képen, így a felhasználó számára is jól láthatóvá válik az elemzés végeredménye.

Az tesztelések eredményei után elmondható, hogy a legtöbb szöveti mintán csekély hibaarányal teljesít az algoritmus. A tesztképeken átlagosan 4,6% a meg nem talált mirigyek aránya, míg a hibásan detektált mirigyeké 12,7%. Az előbbieken említett százalékok a 27. ábra értékeiből számolhatók ki. E ponton megemlítendő, hogy minden, a mirigyen kívül eső, különálló komponenst hibásan detektált mirigyként vettünk, habár legtöbbjük, méretük alapján könnyedén küszöbölhető. Eredményeinket a hasonló rendszerekével is összevetettük, és a vizsgáltak közül rendszerünk találta meg legnagyobb hatékonysággal a mirigyeket.

A projekt a színalapú megközelítésre épül. Ennek előnye, hogy a mirigydetektálás invariáns a mirigyek alakjára, helyzetére és méretére, mivel az algoritmusok a színszegmentálás során kizárólag a különböző szövettani struktúrák egymáshoz viszonyított színkülönbségeit veszik figyelembe ezen struktúrák elkülönítésére. Hátránya e megközelítésnek, hogy nem alkalmas az önmagával, vagy a háttérrel összeolvadt mirigyek elkülönítésére. További hátrány, hogy igen bonyolult korrigáló algoritmusok szükségesek a pontos végeredményhez.

Összességében tehát elmondható, hogy az általunk létrehozott program nagy előnye más megvalósításokkal szemben az, hogy nagyobb arányban találja meg a valódi mirigyeket.

Irodalomjegyzék

- [DELA08] Vincent T. DeVita, Jr., Theodore S. Lawrence, Steven A. Rosenberg, Robert A. Weinberg, Ronald A. DePinho: DeVita, Hellman, and Rosenberg's cancer : principles & practice of oncology (8th ed. ed.). Philadelphia, Wolters Kluwer/Lippincott Williams & Wilkins, 2008
- [DEYE12] Cigdem Demir, Bulent Yener: Automated cancer diagnosis based on histopathological images: a systematic survey, In: Academy of electronic media {online} <http://academy.cs.rpiscrews.us/research/pdf/05-09.pdf>, 2012. április 5.
- [EI05] R Eisner et al.: Improving Protein Function Prediction using the Hierarchical Structure of the Gene Ontology, In: IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, 2005. november
- [FI09] Ficsór Levente, Digitális mikroszkópia az emésztőrendszeri szöveti minták diagnosztikájában, doktori értekezés, SOTE, Klinikai Orvostudományok Doktori Iskola, 2009
- [FIPE03] R. Fisher, S. Perkins, A. Walker and E. Wolfart: Connected Component Labeling {online} <http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm>, 2003
- [GOWO03] Rafael C. Gonzalez, Richard E. Woods, Steve L. Eddins: Digital Image Processing Using MATLAB. Pearson Prentice Hall, 2003
- [HA87] G. Hartmann: Recognition of Hierarchically Encoded Images by Technical and Biological Systems, In: Biological Cybernetics, 1987
- [KA08] Melih Kandemir: Segmentation of Colon Glands by Object Graphs, Bilkent University, Ankara, Törökország, 2008. július
- [KU97] Wolfgang Kühnel (1997), SH atlasz. Budapest, Springer Hungarica Kiadó Kft.
- [LU04] Z. Lu et al.: Predicting Subcellular Localization of Proteins using Machine-Learned Classifiers, In: Bioinformatics, Volume 20, Issue 4, 200, pp. 547 - 556., 2004. március
- [MABE09] Markowitz, S. D., Bertagnolli, M. M.: Molecular basis of colorectal cancer. N. Engl. J. Med. 361 (25): 2449–60, 2009
- [MAC67] J. B. MacQueen: Some methods for classification and analysis of multivariate observations, In: Proc. of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, volume 1, University of California, 1967
- [PATO12] A patológus munkája és szerepe, In: pathology.hu, {online} <http://www.pathology.hu/a-patologus-munkaja-es-szerepe>, 2012. április 4.
- [PRST12] Lutz Priese, Patrick Sturm: Introduction to the Color Structure Code and its Implementation, In: Universität Koblenz-Landau honlapja {online} <http://www.unikoblenzlandau.de/koblenz/fb4/institute/icv/agprieese/research/ColorImgSeg/download/csc.pdf>, 2012. április 5.

- [REM11] Reményi A., Szénási S., Bándi I., Vámosy Z., Valcz G., Bogdanov P., Sergyán S., Kozlovsky M.: Parallel biomedical image processing with GPGPU-s in cancer research 3rd IEEE International Symposium on Logistics and Industrial Informatics (LINDI 2011), Budapest, 2011.08.25-2011.08.27., pp. 245-248. (ISBN:978-1-4577-1840-3)
- [SCCO87] Michael W. Schwarz, William B. Cowan, John C. Beatty: An experimental comparison of RGB, YIQ, LAB, HSV, and opponent color models, In: ACM Transactions on Graphics (TOG) Volume 6 Issue 2, 1987
- [VISO91] L. Vincent, P. Soille: Watersheds in digital spaces: An efficient algorithm based on immersion simulations, In: IEEE T. Pattern Anal. 13, 1991
- [WAX88] B. M. Waxman: Routing of multipoint connections, In: IEEE J. Sel. Area Commun., 6 1617-1622., 1988
- [WEWO98] B. Weyn, G. Van de Wouwer, A. Van Daele, P. Scheunders, D. Van Dyck, E. Van Marck, W. Jacob: Automated breast tumor diagnosis and grading based on wavelet chromatin texture description, In: Cytometry 33, 1998

Függelék

1 Vastagbél mirigyek szegmentálása objektum gráf használatával – részletes kifejtés

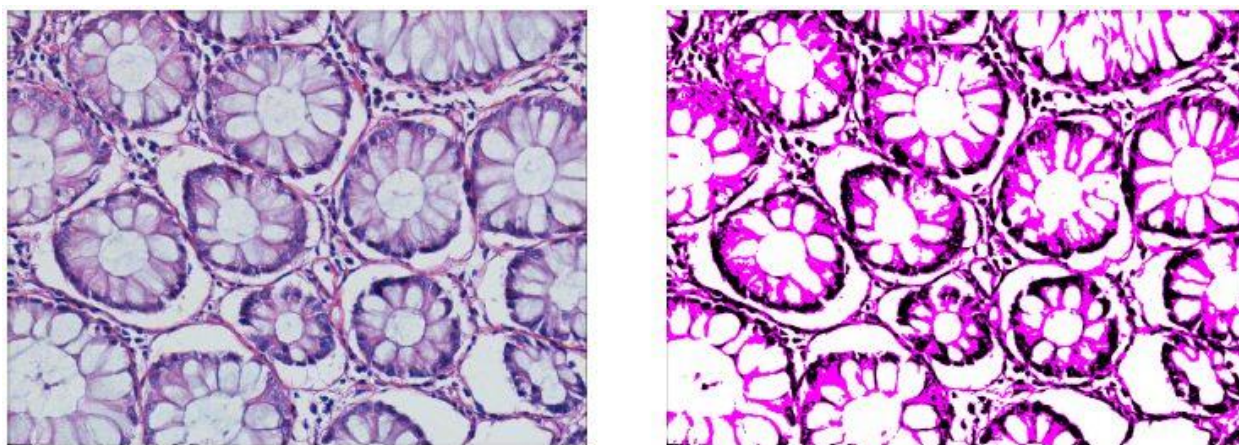
A Függelékben [KA08] részleteit mutatjuk be.

1.1 Pixelek osztályozása

Három lényeges színcsoportba sorolhatjuk a hematoxilin és eozin által festett szöveti képeket (rózsaszín, kékeslila és fehér). A festés szerkezeti felépítése információt ad arról, hogy amely színcsoport, amely szövettani struktúrának felel meg. Például a sejtmagok chromatinban gazdagok, tehát nagy valószínűséggel a sötétlila pixelek a sejtmagban gazdag régiókhoz tartoznak. Emellett a stróma sejtek citoplazmatikus régiói és az összekötő szövetek pedig eozinophilis részek, tehát ezek biztosan rózsaszínűek lesznek. A felhám sejtek citoplazmái és a lumenális részek világosrózsaszínek fehérek, nem úgy, mint a sötétrózsaszínes stróma sejtek, mivel ezek kiválasztó anyagot tartalmaznak, amelynek sem hemotoxylines, sem eozines hatása nincs.

A megmaradt üres régiók, amelyek nem tartoznak semelyik szövettani struktúrába szintelenül maradnak, pontosabban a háttérrel megegyező fehér színűek lesznek.

Felhasználva a korábban felismert, festésből adódó információkat, a pixelek osztályozásához feltételezzük, hogy azok három különálló klaszterbe tartoznak a színtérben. A következő lépés a k-közép klaszterező algoritmus alkalmazása az eredeti pixeladatokon $k = 3$ értékkel, majd a végeredmény ábrázolása a LAB színtérben. A színvektorokat a klaszterekhez kapcsolja olyan módon, hogy azok fehér, rózsaszín és sötétlila színűk által meghatározott súlypontjaik közti euklideszi távolságot veszi figyelembe. Az osztályozás eredménye az 31. ábrán látható.



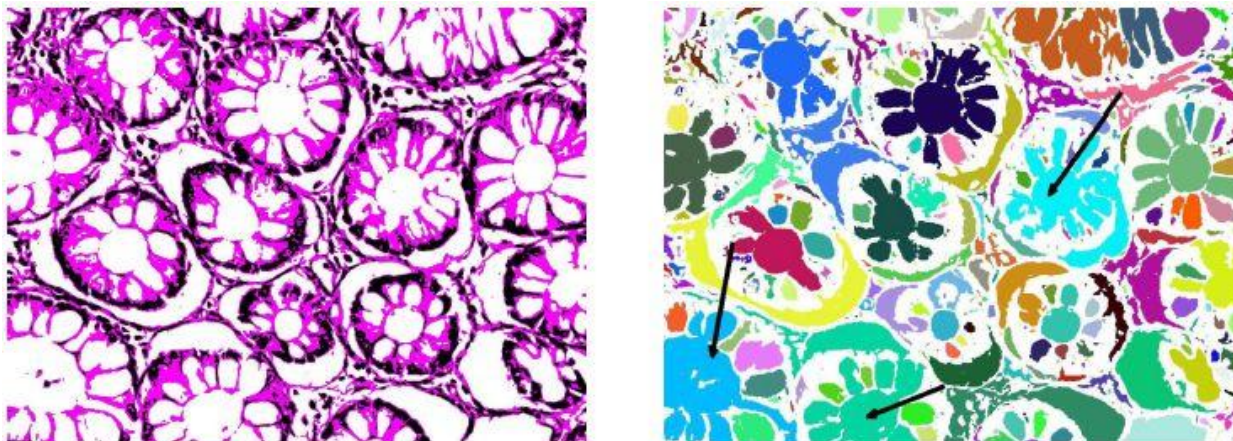
31. ábra: Bal oldalt az eredeti kép, jobb oldalt k-közép algoritmus utáni kép látható. ($k=3$) [KA08]

1.2 Circle-Fit transzformáció

Az osztályozás alapszinten elkülönítette a pixeleket, de ez még nem elég a pontos mirigydetektáláshoz. A magrégiók ismertek, de nem lehet egyenként behatárolni őket. Ha összefüggő komponens analízist hajtunk végre a magként megjelölt pixeleken, akkor az nem ad kellően megbízható eredményt, mivel a felhám sejtmagjai néhol nagyon közel állnak egymáshoz és így kialakul belőlük egy összefüggő komponens. Egy magasabb szintű képpontszeparációra van szükség. További probléma, hogy a lila pixel tartozhat felhám,- és strómasejthez is. Ugyanígy tartozhat fehér pixel egy felhám citoplazmához vagy egy luminális részhez, vagy tartozhat az üresnek minősített régióhoz is. Jól látható, hogy kizárólagosan a pixel-alapú információk nem elegendőek, a probléma megoldásához. Ezért érdemes objektumokat létrehozni minden egyes szövettani struktúrához, és az ezekből kinyert információkkal dolgozni ahelyett, hogy önmagában a pixel-alapú információkat használnánk.

Az objektumok meghatározásához ideális út az, hogy a szövettani egységek közvetlen szegmentálása után az összefüggő komponensek lesznek objektumként definiálva. Más szavakkal, ez a mirigyszegmentáció kisebb alproblémákra bontása. Külön érdemes kezelni a sejtmag, a luminális részek és a felhám citoplasma szegmentációt majd, a végeredményeket összegezve a mirigy körvonalai az eredmény. Nincs előző, ismert tanulmány, amely a citoplasma és lumen szegmentációra fókuszált volna.

Az objektum definiálás másik módszere az, hogy úgy azonosítunk minden összefüggő komponens a képen, mint különálló objektum. Általánosságban akkor lehet alkalmazni, miután morfológiai műveletekkel a zajok el lettek simítva. A kísérlet azonban azt mutatta, hogy nem alkalmazható ez az eljárás, mivel az objektumok nem eléggé szeparáltak. Például valószínűsíthető, hogy fehér régiók alakulnak ki a mirigy határán, és ha a belső magpixelek nincsenek teljesen elzárva ettől a külső fehér résztől, akkor összeolvadnak, ez a jelenség látható a 32. ábrán. Ez a módszer tehát használhatatlan a magasabb szintű szeparáláshoz.

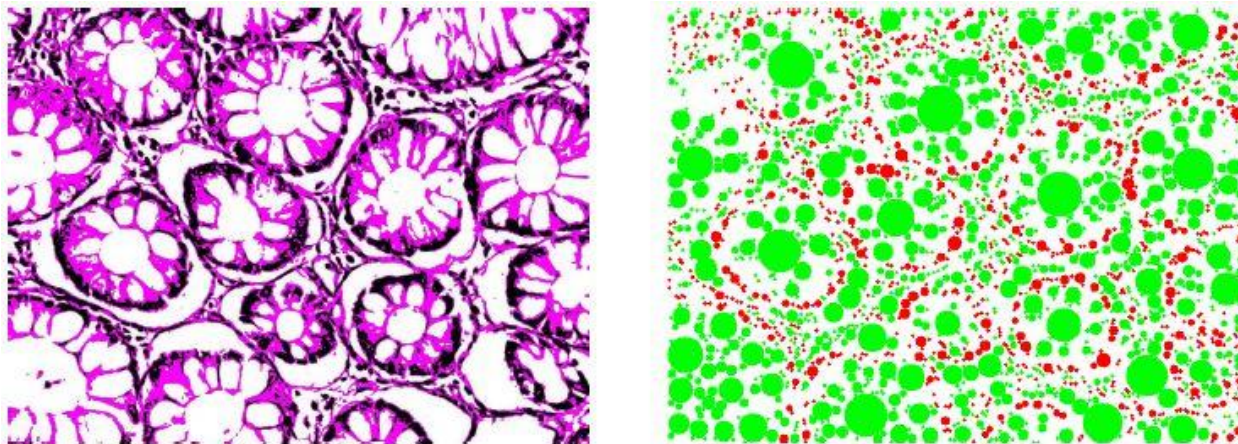


32. ábra A baloldali ábra egy klaszterezett kép, a jobboldali ábra pedig a fehér szín összefüggő komponensei (a nyilak jelzik a hibás összeolvadást) [KA08]

A fent említett nehézségek miatt egy új algoritmus vezetett be az alkotó, a *Circle-Fit* algoritmust, amely a nyers pixel adatokat csoportosítja körkörös objektumokba. Ezentúl ezek az elemek fogják reprezentálni a szövettani struktúrákat.

Az algoritmus kimenete számos objektumot tartalmaz, amelyek közül azonban nem mindegyik kör alakú. Keletkezhetnek félkör alakú objektumok is, ezek akkor fordulnak elő, amikor egy nagyobb sugarú kör, részben felülír egy előzőleg létrehozott kört. Ezek a régiók általában elég kicsik ahhoz, hogy figyelmen kívül lehessen hagyni őket, vagy pedig olyan nagyok, hogy egy körcsoporttal reprezentálhatóak. Azért, hogy ezen problémás esetek le legyenek kezelve, ismét új algoritmus vezetett be a készítő, az *Iterative Double Circle-Fit*, egy olyan speciális *Circle-Fit*, amely iteratívan hajtódik végre és a kimenetei csak körcsoport objektumok lehetnek. Az algoritmus az utófeldolgozásban elhagyja a kicsi és nem körkörös objektumokat, ahol a „kis méretet” egy küszöb segítségével szűri, a körkörösséget pedig egy kerekesség vizsgáló függvény végzi (33. ábra).

Azért kell a kör alakú elemeket vizsgálni, mert a feldolgozás szempontjából minden fontos szövettani egység kör alakú. A sejtmagok általában körrel leírhatóak, a luminális területek egy, vagy néhány nagy körként jelennek meg a mirigyek közepén.



33. ábra: A baloldali ábra a klaszterizált kép, a jobboldali kép pedig a double circle-fit algoritmus futtatása után. A fekete klaszter köreit piros, a fehér klaszter köreit zöld szín jelöli [KA08].

1.3 Mirigyjelöltek detektálása

A következő lépés azon területek megtalálása, amelyek valószínűleg mirigyekhez tartoznak. Elsőként a mirigymagok detektálása megy végbe, aminek elsődleges célja megtalálni azon pixelhalmazokat, amelyek a mirigytesten belül helyezkednek el, majd ezt követően régiónöveléssel lehet megállapítani a mirigyek határvonalait az előbb felismert magból kiindulva.

Az Iteratív Double Circle-Fit transzformáció által meghatározott körcsoportok súlypontjából indul a régiónövelés, és így alakul ki egy mirigy jelölt. Ebből a halmazból, valódi mirigyeket a hibásan detektált mirigy elkülönítési lépésben határozzuk meg.

1.3.1 A hibásan detektált sejtmag felismerése és eltüntetése

A k-közép klaszterizáció után a létrejött kép fehér és fekete összefüggő komponenseket tartalmaz. A Circle-Fit algoritmus két kimeneti képet hoz létre (egyet a fekete és egyet a fehér komponenseknek). A továbbiakban a komponenseket fehér és fekete köröknek nevezzük.

Az eredeti képen a fehér körök megfelelnek a luminális részeknek, a felhámsejt citoplazmájának vagy az üres régióknak. Ezen szövettani struktúrák közül az első kettő a mirigy belsejében helyezkedik el, így ezek elkülönítése nem szükségeltetik. Ellenben fontos lesz a harmadik (üres régió) elkülönítése a többitől, amelynek bővebb kifejtése a *Mirigymag detektálás* és a *Hibásan detektált mirigyek elkülönítése* fejezetben található.

A fekete körök lehetnek sejtmag vagy sötétebb zajos régiók. A hibásan detektált sejtmag felismerés érdekében, el kell különíteni ezeket egymástól, majd a zajos részeket képviselő köröket el kell hagyni. A zajos részek meghagyása meggátolná a folyamatos mirigykörvonal felismerését, és az eredmény részleges, azaz nem összefüggő határvonal lenne.

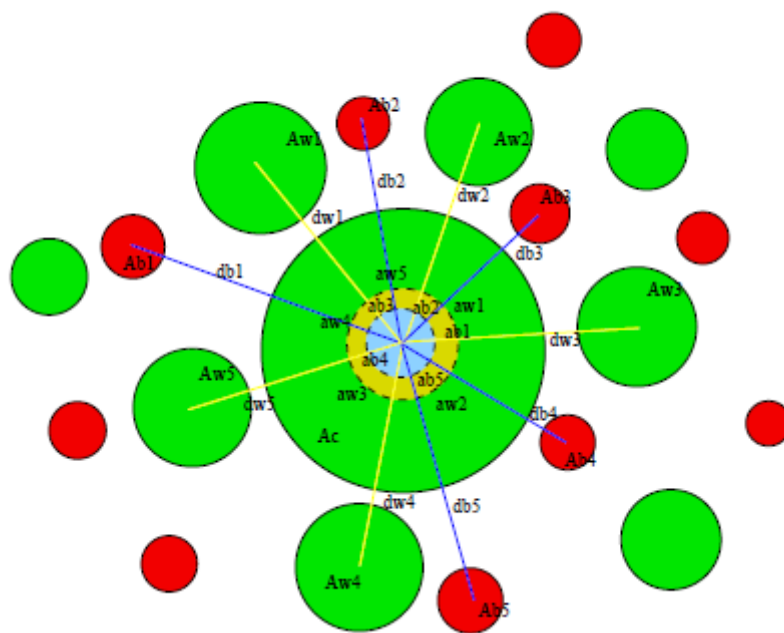
1.3.2 Mirigymag detektálás

Ebben a lépésben azon mirigymaghoz tartozó képpontokat keressük, amelyekből a későbbiekben a régiónövelés elindulhat. Kijelölünk egy olyan pixelt, amely közel van az aktuálisan vizsgált mirigy súlypontjához.

Mivel a kép objektum-alapú reprezentációjával történik a feldolgozás, ezért egy új mirigymag detektálást dolgozott ki a készítő, amely kihasználja az ilyen formájú leírást. A képen található fehér körök két klaszterbe különülnek jellemzőik alapján, a k-közép algoritmus használatával. Az egyik csoport a luminális területeket, illetve a sejtcitoplazmákat, a másik csoport pedig az üres, fehér régiókat reprezentálja. Egy fehér körnél az alábbi jellemzőket kell vizsgálni:

1. A vizsgált kör súlypontja és k legközelebbi fekete kör szomszédainak súlypontjától mért távolságok
2. A vizsgált kör súlypontja és k legközelebbi fehér kör szomszédainak súlypontjától mért távolságok
3. A vizsgált kör súlypontjából k legközelebbi fekete kör szomszédainak súlypontjáig húzódó vonalszegmensek polár szögei
4. A vizsgált kör súlypontjából k legközelebbi fehér kör szomszédainak súlypontjáig húzódó vonalszegmensek polár szögei
5. A vizsgált kör legközelebbi fekete kör szomszédainak területe
6. A vizsgált kör legközelebbi fehér kör szomszédainak területe
7. A vizsgált fehér kör területe

A fenti felsorolás vizuális ábrázolása a 34. ábrán látható. Azért vizsgálják ezen jellemzőket, mert a fehér körök legközelebbi körszomszédjainak térbeli elhelyezkedése más lesz a mirigy területeken belül, mint a mirigy területeken kívül. Az aktuálisan figyelt elemhez legközelebb eső fekete körök várhatóan a felhámon lévő sejtmagoknak felelnek meg. Ezen sejtmagoknak igen jellegzetes az elhelyezkedésük, ugyanis ha mirigy köröket vizsgálunk, akkor általában csak az egyik oldalon lesznek találhatóak, azonban ha nem-mirigy területről van szó, akkor pedig homogén eloszlás figyelhető meg, az aktuálisan vizsgált kör körül. Ez az elhelyezkedés a polárszögek elváltozásaiban figyelhető meg. Ezen elül, a felhámon elhelyezkedő sejtmagok általában nagyobbak, mint a stroma sejtmagok.

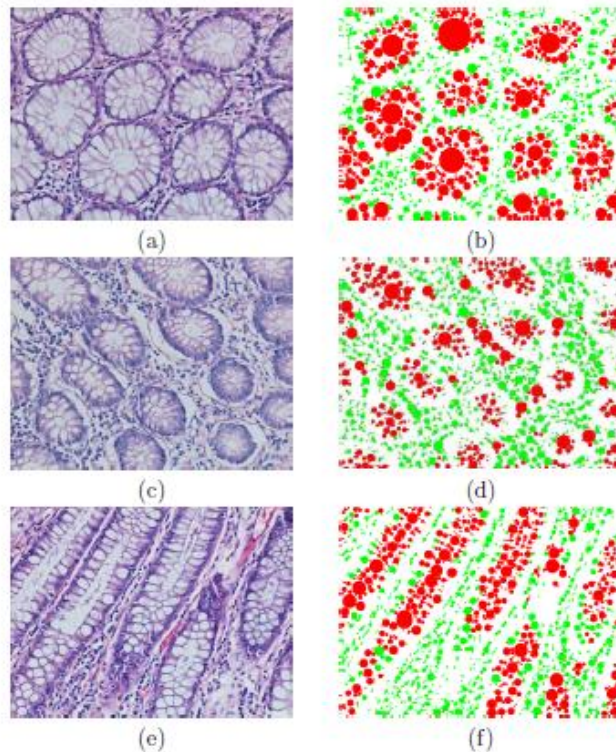


34. ábra:

Aw1-5: az 5 legközelebbi fehér kör területe
 Ab1-5: az 5 legközelebbi fekete kör területe
 aw1-5: az 5 legközelebbi fehér körös polár szögek
 ab1-5: az 5 legközelebbi fekete körös polár szögek
 dw1-5: távolság a vizsgált kör és az 5 legközelebbi fehér kör közt
 db1-5: távolság a vizsgált kör és az 5 legközelebbi fekete kör közt
 [KA08]

A fehér körök klaszterizálása után, a mirigy és nem-mirigy köröket a következőképpen lesznek meghatározva: minden klaszterben kiszámításra kerül az átlagos átmérő, a nagyobb méretűek lesznek a mirigy, a kisebbek pedig a nem-mirigy csoportok.

A fenti algoritmus eredményei a 35. ábrán láthatóak. Az (a) részkep egyenletes, körkörös formájú mirigyeket tartalmaz, amelyek detektálása a legegyszerűbb (b). A (c) részképen már kisebbek a mirigyek, és a bennük lévő sejt citoplazmák és a luminális részek észrevehetősége igen nehéz, és ez az eredményen is meglátszik (d). Az utolsó részképen (d) a mirigyek már teljesen más alakban helyezkednek el, ez szintén nehezíti a detektálást (e).



35. ábra

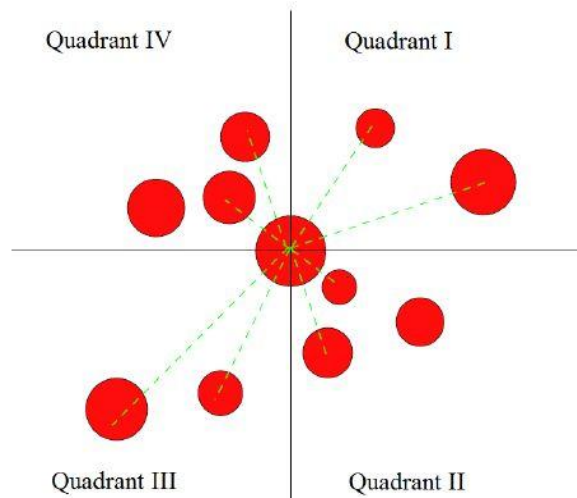
Három szöveti minta és az azon levő fehér körök osztályozása utáni eredmény. A piros körök jelölik a mirigy, a zöld körök pedig a nem-mirigy területeket [KA08].

1.3.3 Régiönövelés

Miután a mirigyek magjaihoz tartozó képpontok megvannak, a következő lépés a határvonalak megtalálása lesz az alábbi mirigyjellegzetességek alapján:

- Egy mirigy körül felhámsejtek találhatóak. Az *Iterativ Double Circle-fit* transzformált fekete klaszterben kell, hogy legyen egy körlánc, amely a mirigy határán helyezkedik el. A körlánc elemei nagyon közel helyezkednek el egymáshoz.
- Nincs sejtmag a mirigyen belül. A hamis magelkülönítés után, ezen régiók várhatóan üresek.

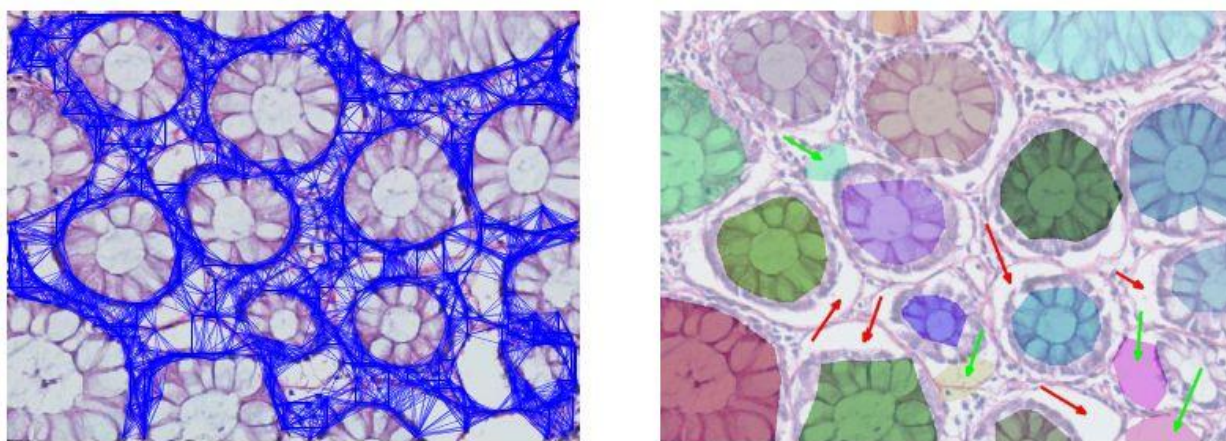
E tulajdonságok segítenek megtalálni, azon elválasztó részeket, ahol a régiönövelésnek meg kell állnia. Készítünk egy gráfot a fekete körökből. A pontok jelentik a kör súlypontját, az élek pedig egypixel széles vonalakat, amelyek egy kör súlypontjától egy másikig tartanak. A fekete körök éleit a következők jelentik: a képet négy negyedre bontjuk (36. ábra). A kör minden negyedben a *K legközelebbi* szomszédjaihoz kapcsolódik, amennyiben ezek a körök *N legközelebbi* szomszédságában vannak. A '*K*'-t úgy definiáljuk, mint négyes szomszédság és '*N*'-t, mint teljes szomszédság. Az algoritmus egy adott bináris képen létrehozza a sejtmag hálózatot. A kimeneti képet azon maghoz kapcsolódó képpontokkal lehet használni, amelyek az előző lépésben lettek meghatározva, a mirigyek belső széleinek keresésekor.



36. ábra: A kör közepe és a négy negyed [KA08]

Az eredményt a 37. ábra, bal részképe mutatja. Látható, hogy nincs olyan él, amely keresztülhaladna a mirigyként felismert régiókon. Ezen régiók olyan összefüggő komponensek, amelyek régiónöveléssel nyerhetők ki, amely növelés az előző lépésben detektált magpixelekből indul ki, és egészen a szélekig tart. Megfigyelhető, hogy van olyan üres régió (piros nyíl jelöli a 37. ábra jobb oldali részképén), amely nem mirigyekhez tartozik. Mivel a fehér körök ehhez tartoznak, ezért eszerint is tároltuk el őket (mint nem-mirigy régiók).

A megjelölt mirigyek között található néhány, valójában üres régió (zöld nyilak jelzik a 37. ábrán), tehát szükséges egy szabálygyűjtemény az ilyen esetek kiküszöbölésére.



37. ábra: A régió növelés eredménye (bal oldal) és az eredeti képre helyezett eredmény (jobb oldal) [KA08]

1.4 Hibásan detektált mirigyek elkülönítése

Az előző lépésben helytelenül detektált mirigyek „téves riasztásnak” tekinthetőek, amelyeket el kell távolítani. Az elkülönítés problémájára egy felügyelt tanuló rendszert terveztek.

A rendszer működéséhez meg kell számolni a fehér, fekete és rózsaszín képpontokat, továbbá, hogy ez a szám hány százaléka az egész területnek a mirigy régió-, a konvex burkon-, a k-pixel széles dilatációs övön belül.

A tanító mintakészlet a fent említett jellemzők alapján készül el és manuálisan megjelölésre kerül az, hogy egy mirigyjelölt ténylegesen mirigy-e, vagy sem. E mintakészlet segítségével tanítható be az a döntési fa, amelyre egy szabályrendszert alkalmazva, elkülöníthetők a hibásan detektált mirigyek.

A szabályrendszer 5 pontból áll, amely mindegyike valamilyen előre definiált küszöbértékhez viszonyítja a fent említett jellemzőket, és ez alapján hoz döntést.

1.5 Mirigyhatárvonalak felismerése

Az utolsó lépés a határvonalak meghatározása, amelyben először egy körkörös elemmel dilatálunk. Ezekután a dilatált régiókon belül megtalálhatóak a fekete körök. E körök sorba vannak rendezve a fekete körök súlypontjától a régiók súlypontjáig tartó vonal és az x tengely által bezárt szög nagysága szerint. Az így létrejött rendezett pontcsoportok egyszerű polinomoknak felelnek meg. A poligonok csúcsait összekötjük N-szomszédság szerint, ahol N adja meg a kirajzolás pontosságát. Ha $N=1$, a kimeneti kép egy nagyon egyszerű sokszög lesz. Ha viszont N megegyezik a régióban található sejtmagok számával, akkor megkapjuk a poligon legrészletesebb alakját (38. ábra).



38. ábra: A végeredmény a kezdeti szöveti képre helyezve [KA08]