

**Talajminták látható szerkezetének vizsgálata és
kiértékelése számítógépes program segítségével**

SZAKDOLGOZAT

Szerző: Hovánszki György

email: hovanszki@freemail.hu

Külső konzulens: Dr. Braun Mihály

2001. Debrecen

Köszönet

Köszönetet szeretnék mondani mindazoknak, akik építő jellegű megjegyzéseikkel segítettek a munkámat, illetve válaszoltak a sokszor lehetetlen kérdéseimre.

Külön köszönet Dr. Braun Mihálynak, aki az alapötletet adta a program megírásához, és mindvégig tanácsokkal látott el.

Köszönöm még Kecskeméthy Dóra a türelmét, aki az angol hiányosságaimat pótolta.

Nem utolsó sorban Pere Zsoltnak, aki ellenőrizte a munkámat.

Köszönetet szeretnék még mondani mindazoknak, akik a SediGraph programot használni fogják a jövőben, elismerve ezzel a tevékenységemet.

Nyilatkozat

Ezennel tudatom teljes birtokában kijelentem, hogy a SediGraph szoftver az én saját szellemi termékem, és nincs tudomásom hasonló nevű programról.

A programot a NET-INFO kft. Számítógépes Programozó tanfolyamon végzett tanulmányaim lezárásaként szakdolgozatnak szánom, ami emellett a Debreceni Egyetem Szervetlen és Analitikai Kémiai Tanszék munkáját segíti.

A programot jövőbeni terveim szerint egy angol szakmai folyóiratban szeretném publikálni, ezért angol menürendszerrel írtam meg. Idő hiányában sajnos ezt már nem tudtam magyarra átírni, de kárpoztulás a teljes sűgő rendszer magyar nyelvű lett.

Tartalomjegyzék

1	BEVEZETÉS.....	7
2	A RENDSZER ÁLTALÁNOS LEÍRÁSA.....	10
3	A FUTTATÁSHOZ SZÜKSÉGES HARDWARE-SOFTWARE KÖRNYEZET.....	11
4	A PROGRAM RÉSZLETES ISMERTETÉSE PROGRAMOZÓI SZEMMEL	12
4.1	MEGNYITÁS – OPENDIALOG.....	13
4.2	MENTÉS.....	15
4.3	NYOMTATÁS.....	16
4.4	A KÉP.....	17
4.5	A SZÁMOLÁS	17
4.6	A GRAFIKON	18
4.7	A MENÜ.....	19
4.8	AZ ESZKÖZTÁR.....	19
4.9	AZ INDEX MENTÉSE.....	20
4.10	AZ INDEX MEGNYITÁSA.....	21
4.11	A GYORSSÚGÓ	21
4.12	SÚGÓ – HELP	22
5	A SZÁMOLÁS ELJÁRÁS KIÉRTÉKELÉSE	24
5.1	AUTO CUTTING PICTURE.....	24
5.2	AZ ÁTLAG (SZÁMTANI KÖZÉP) KISZÁMOLÁSA.....	25
5.3	A RELATÍV SZÓRÁS KISZÁMOLÁSA (RSD%).....	26
5.4	A MEDIÁN ÉRTÉK SZÁMOLÁSA.....	27
5.5	A HISZTOGRAM.....	28
5.6	A HORIZONTAL AVERAGE.....	28
5.7	AZ RGB SZÁMOLÁS.....	29
5.8	A SMOOTH KISZÁMÍTÁSA	30
5.9	A DERIVE KISZÁMÍTÁSA.....	31
6	AZ ADATMODELL	32
6.1	ELŐKÉSZÜLETEK.....	32
6.2	BEMENETI ADATOK.....	32
6.3	INDEX FILE.....	33
6.4	KIMENETI ADATOK	34
6.5	ADAT FILE.....	34
7	A PROGRAM ISMERTETÉSE FELHASZNÁLÓI SZEMMEL	36
7.1	A SEDIGRAPH PROGRAM TELEPÍTÉSE	36
7.2	A PROGRAM FELÉPÍTÉSE	37
7.2.1	<i>Menü.....</i>	<i>38</i>
7.2.2	<i>Az eszköztár (ToolBaar).....</i>	<i>38</i>
7.2.3	<i>A grafikon (Chart).....</i>	<i>38</i>
7.2.4	<i>A kép (Picture)</i>	<i>39</i>
7.2.5	<i>A gyorsítógé.....</i>	<i>39</i>
8	MENÜRENDSZEREK TÁRGYALÁSA FELHASZNÁLÓKNAK	40
8.1	A MENÜPONTOK	40
8.2	FILE	40
8.2.1	<i>Open.....</i>	<i>40</i>
8.2.2	<i>Save Chart.....</i>	<i>41</i>
8.2.3	<i>Save Data</i>	<i>41</i>
8.2.4	<i>Save Picture</i>	<i>41</i>
8.2.5	<i>Print</i>	<i>41</i>

8.2.6	<i>Exit</i>	42
8.3	VIEW	42
8.3.1	<i>ToolBaar</i>	42
8.3.2	<i>Picture</i>	42
8.4	CHART	43
8.4.1	<i>Average</i>	43
8.4.2	<i>Median</i>	43
8.4.3	<i>Dispersion</i>	43
8.4.4	<i>RGB</i>	43
8.4.5	<i>Derive</i>	44
8.4.6	<i>Hisztogram</i>	44
8.4.7	<i>Horizontal Average</i>	44
8.4.8	<i>Smoothing</i>	45
8.5	SETTING	45
8.5.1	<i>Auto Cutting Picture</i>	45
8.5.2	<i>Horizontally Reverse</i>	45
8.5.3	<i>3 Dimmension Chart</i>	46
8.6	INDEX	46
8.6.1	<i>Make Index</i>	46
8.6.2	<i>Open Index</i>	46
8.7	HELP	47
8.7.1	<i>Contents</i>	47
8.7.2	<i>Search for help on</i>	47
8.7.3	<i>About</i>	47
9	A SEDIGRAPH PROGRAM ISMERTETÉSE PÉLDÁN KERESZTÜL	48
9.1	AZ INDEX FÁJL KÉSZÍTÉSE KÉPEKHEZ	53
9.2	INDEX FÁJLOK FELDOLGOZÁSA:	54
9.3	TÖBB KÉP ÖSSZEFÜZÉSE EGY GRAFIKONBA	54
9.4	AZ ADATOK ELMENTÉSE:	56
9.5	AZ ÁTLAG GÖRBE SIMÍTÁSA	57
10	A SEDIGRAPH PROGRAM ELTÁVOLÍTÁSA.....	58
11	A PROGRAM TESZTELÉSE	59
11.1	A SEBESSÉG MÉRÉSE:	59
11.2	A FELBONTÁS VIZSGÁLATA	61
12	ÖSSZEGZÉS	63
13	IRODALOMJEGYZÉK:	64

Ábrák

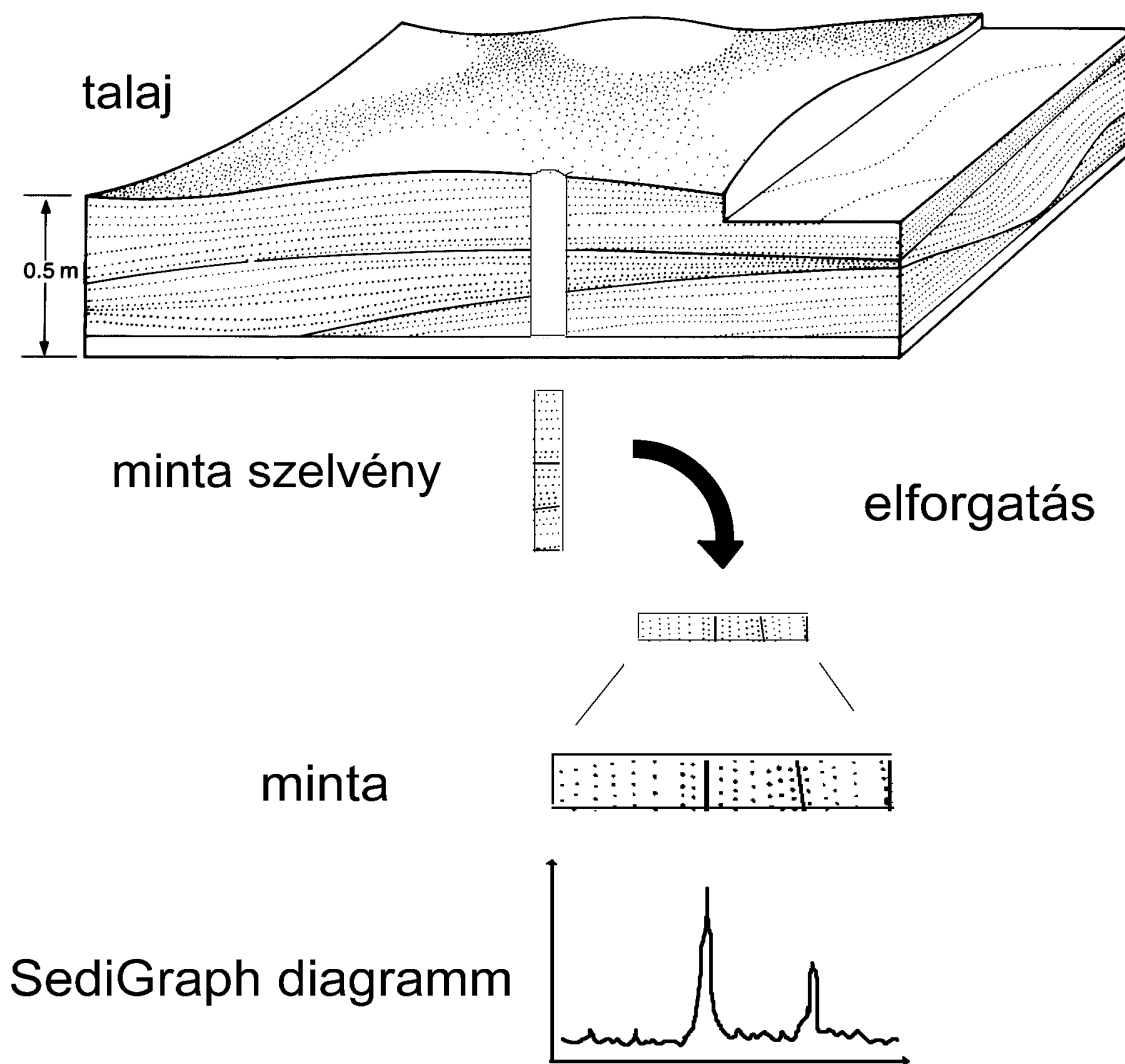
1. Ábra: Az alkalmazott objektumok kapcsolatainak rendszere.....	12
2. Ábra: Az operációs rendszer beépített mentés ablakát használja.....	15
3. Ábra: A nyomtatás beállításai egy külön ablakban történnek.....	16
4. Ábra: A programban használt eszköztár.....	19
5. Ábra: Az index fájl készítésére használt űrlap.....	20
6. Ábra: A gyorsító mindig az aktuális információkat mutatja.....	21
7. Ábra: A SediGraph program súgó rendszerének a tartalomjegyzéke.....	22
8. Ábra: A telepítés során megjelenő ablak.....	36
9. Ábra: A Sedigraph program főablaka.....	37
10. Ábra: A Sedigraph menürendszere.....	40
11. Ábra: A megnyitás ablak képe.....	48
12. Ábra: A program nézeti képe az átlag (average) görbével.....	49
13. Ábra: Az átlag és a derivált görbe.....	50
14. Ábra: A színes kép RGB görbéinek a megjelenítése. A színátmenetknél a görbék helyet cserélnek.....	51
15. Ábra: A képen egy kitakart minta képe és átlag görbéje látható.....	52
16. Ábra: A Make Index segédablak nézeti képe.....	53
17. Ábra: Több kép összekapcsolására szolgáló ablak. A képekhez adjuk meg a kalibrációt.....	55
18. Ábra: A Sedigraph program által készített adatok feldolgozása EXCEL táblázatkezelővel.....	56
19. Ábra: A simított görbe illeszkedése az átlag görbére. Lent a számolás útján előállított kép látható.....	57
20. Ábra: A sebességek grafikus ábrázolása.....	60
21. Ábra: Felbontás vizsgálata derivált görbével. 100% = 1200 x 400 pixel.....	61
22. Ábra: Felbontás vizsgálata derivált görbével. 50% = 600 x 200 pixel.....	61
23. Ábra: Felbontás vizsgálata derivált görbével. 25% = 300 x 100 pixel.....	62
24. Ábra: Felbontás vizsgálata derivált görbével. 10% = 120 x 40 pixel.....	62

Táblázatok

1. Táblázat: Az adatok feldolgozásának eljárásai.....	8
2. Táblázat: Mért sebességi adatok a SediGraph programmal.....	60
3. Táblázat: A felbontás vizsgálatánál alkalmazott képméretetek.....	61

1 Bevezetés

Vegyész diplomamunkámat a KLTE Szervetlen és Analitikai Kémiai Tanszékén írtam. Itt kerültem kapcsolatba üledékek, illetve talajszelvények vizsgálatával. A mintákat függőlegesen veszik a talajból; holtágak esetén például minden áradáskor újabb üledékréteg képződik, amelyeknek egy méteres hossza akár több száz éves is lehet. Az üledék egyes rétegei - attól függően, hogy honnan származik a lerakódás anyaga - változó összetételűek és színűek lehetnek (homok: sárgás, agyag: fekete, mészkő: fehér, stb.). Ezek, mivel nem egyforma színűek, szemmel láthatóan réteges szerkezetet hoznak létre, ami adott helyre jellemző lehet [1].



A mérések legtöbbször az elemösszetételre terjednek ki, de a szemmel látható struktúrának ugyancsak információhordozó szerepe lehet [2].

A programom ezt az információt szeretné feldolgozni, és kiértékelhető adatokká alakítani.

A program a talajszelvények képét vertikálisan dolgozza fel, és ábrázolható, kiértékelhető és összehasonlítható adatokat eredményez. Az adatok feldolgozása többféleképpen történik (1. táblázat).

1. Táblázat: Az adatok feldolgozásának eljárásai.

eljárások	magyarázat
Átlag	a pontok átlagát adja vissza
Medián	a pontok mediánját adja vissza
Szórás	a pontok szórásszázalékát adja vissza
Vörös színtkomponens	a vörös színtkomponens mennyiségét adja
Zöld színtkomponens	a zöld színtkomponens mennyiségét adja
Kék színtkomponens	a kék színtkomponens mennyiségét adja
Hisztogram	a összes pontból számolt hisztogram
Horizontális ábrázolás	a pontok horizontális átlagát adja vissza
Simítás	az átlag pontokat simítja egy 5 pontos mozgóátlaggal
Deriválás	az átlag pontok deriváltját veszi mozgópontosan

A számolt eredményeket grafikusán ábrázolja, de az adatokat el is menthetjük, így tetszőleges táblázatkezelővel (pl. EXCEL) azok a programtól függetlenül is feldolgozhatók. A mintavétel hossza korlátozott (minta tömörödése, szállíthatóság, sérülékenység), ezért egyes esetekben átfedéssel dolgoznak; ekkor a vizsgálandó mintát több képre bontják. A program a munkánkat megkönnyítve ezekből a részekből önállóan összeállítja a teljes fúrásmintát, illetve segít megkeresni az azonos részeket.

A program nem csak talajvizsgálatra képes, hanem minden olyan objektum vizsgálatára, aminek réteges, látható tulajdonsága van; például fák kormeghatározására is használható. Az évente ismétlődő fagyűrűk távolsága az időjárás változékonyságától függően különböző lehet. Tehát, ha egy régebben kidőlt fa erezetének a programmal előállított grafikonját egy még most is élő fa grafikonjával összehasonlítjuk, akkor meg kell azt találnunk benne, feltéve, ha a fa legalább annyi idős, mint a kidőlt.

Kezdetben speciálisan a talajvizsgálatra összpontosítottunk. ám fölfedezvén azt, hogy milyen kevés olyan program van, ami ilyen témakört dolgoz fel, általánosítottuk, és kiterjesztettük a használhatóságát, nagy szabadságot adva a felhasználónak; ugyanakkor egyes részei kimondottan a talaj vizsgálatát célozzák meg.

2 A rendszer általános leírása

A program 32bit-es Windows operációs rendszerre készült.

Előnyei:

- Elterjedt operációs rendszer.
- Könnyen kezelhető programok.
- Azonos programkörnyezet (már megszokott gombok, kapcsolók, menük, súgók).
- A forráskép formátuma elterjedt (BMP.)
- A nyomtatót nem kell külön installálni.
- A súgó használata egyszerű, könnyen kezelhető.

Hátránya:

- Hardverigényes

A program minden olyan gépen képes futni, amelyen 32 bites Windows operációs rendszer van feltelepítve. Kivétel a monitor, mert itt legalább 256 színmélység szükséges a helyes megjelenítéshez, és 800x600 felbontás ajánlott. Minél nagyobb a képernyő felbontása, annál jobb ábrázolás lehetséges (esetenként több ezer pontból álló grafikont is használ a program).

A program CD lemezen van, amelyről telepíteni kell, ezért a telepítéshez CD-ROM olvasó is szükséges.

A program telepítéséhez 1,5 MB üres hely szükséges a merevlemezen. A program telepítéskor létrehoz egy minta könyvtárat (Sample Directory) is, amelybe 5 db példa képet is elhelyez.

A program által használt állományok:

SediGraph.exe	<i>a program fő feldolgozó része</i>
SediGraph.hlp	<i>a program súgó részét tartalmazza</i>
SediGraph.cnt	<i>a súgó tartalomjegyzékét tartalmazza</i>

3 A futtatáshoz szükséges hardware-software környezet

Minimális hardware követelmény:

- PC-AT 100 MHz
- 16 MB memória
- merevlemez (legalább 2 MB szabad területtel)
- monitor (színes)
- videokártya (legalább 256 szín)
- CD-ROM meghajtó
- billentyűzet
- egér
- szkennер

Ajánlott hardware követelmény:

- PC-AT 200 MHz
- 32 MB memória
- merevlemez (5 MB szabad területtel)
- monitor (színes, „15”)
- videokártya (16 bit színmélység)
- CD-ROM meghajtó
- billentyűzet
- egér
- szkennер (600x600 dpi)
- nyomtató (300x300 dpi)

Software követelmények:

- Win9x 32 bites operációs rendszer (Win95, Win98, Windows ME)

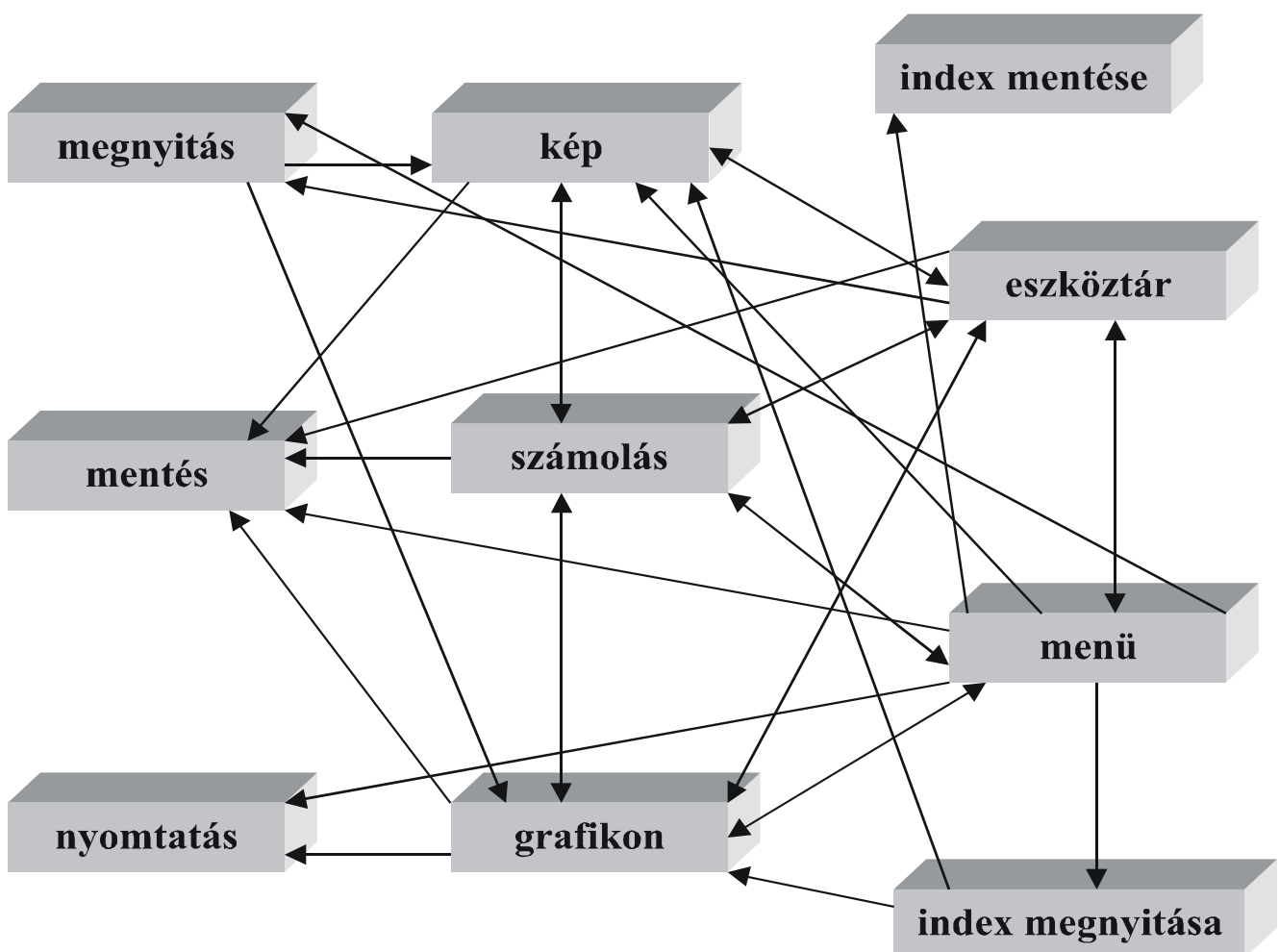
Megjegyzés: A legtöbb 32 bites Windows operációs rendszerű gépen már fut a program!

4 A program részletes ismertetése programozói szemmel

A programot Delphi 5 Objektum Orientált fejlesztői környezetben írtam. Azért döntöttem a Delphi mellett, mert ezzel gyors, esztétikus és jól kezelhető program írható [3]. A készített program 32 bites Windows operációs rendszeren fut. Lefordítottam 16 bitesre és kipróbáltam, de mivel általában itt kisebb teljesítményű gépeket használnak, a sok számolás miatt nagyon lelassult.

A program Objektum Orientáltsága miatt folyamatábrát készíteni hozzá merész vállalkozás lenne, ezért inkább az egyes objektumokról és ezek kapcsolatáról írok részletesebben.

Az 1. ábra a program főbb részeit és ezek kapcsolatát mutatja be,. A nyilak iránya a kommunikáció iránya is egyben. Jól látható, hogy még nagyobb részek ábrázolása esetén is milyen bonyolult hálózat alakul ki, ezért az objektumok nagyobb felbontása a kiértékelés szempontjából indokolatlan.



1. Ábra: Az alkalmazott objektumok kapcsolatainak rendszere.

4.1 MEGNYITÁS – *OpenDialog*

A program bemeneti adatként képet használ. A kép beolvasása a már megszokott általános windows megnyit ablakkal történik. A kép formátuma Windows Bit Map. Azért döntöttem emellett, mert igen elterjedt a windows operációs rendszer alatt, és tömörítetlensége miatt, nincs adatvesztés, s ez fontos szempont. A windows rendszer ismer olyan megnyitó ablakot is, amelyet kimondottan képekre alakítottak ki, azaz beolvasás során már megmutatja kicsiben. Ezt a rendszert is kipróbáltam, de ebben az esetben nem lenne célszerű használni, mivel az itt olvasott képek gyakran igen nagyok, és ezért nagyon lelassítják a programot (akár 20 másodperc is lehetett a késés; ez első ránézésre nem tűnik soknak, de több száz kép esetén már igen jelentős idővesztéssé válik).

Megnyitás után a program a képet egy KÉP objektumnak adja át.

Az indexek beolvasása is itt történik. Ez egy speciális TEXT fájl, amit a programmal készítünk el. Beolvasáskor keresi az NDX kiterjesztésű fájlokat: megvizsgálja, és ha nem egyezik a formátuma, akkor hibaüzenetet küld.

Rövid forráskód értelmezés:

opendialog ablak címének megadása

```
OpenDialog1.Title:='Open picture';
```

igaz, ha az OpenFileDialog ablak OK-val zár begin ... end ciklusban

```
If OpenDialog1.Execute Then begin
```

aktuális könyvtár átadása más objektumoknak is

```
OpenDialog2.InitialDir:=OpenDialog1.InitialDir;
```

a file átadása a KÉP (IMAGE1) objektumnak

```
Image1.Picture.LoadFromFile(OpenDialog1.FileName);
```

a kép objektom láthatóság tulajdonságának a bekapcsolása

```
Image1.Visible:=True;
```

a kapcsoló gombok megfelelő állapotának beállítása

```
Kapcsol(True);
```

a főablak újrarajzolása

Form1.Refresh;

a számolások meghívása

Szamol.Click;

A GRAFIKON objektum címe legyen a beolvasott file elérési útja és neve

Chart1.Title.Text:=OpenDialog1.Files;

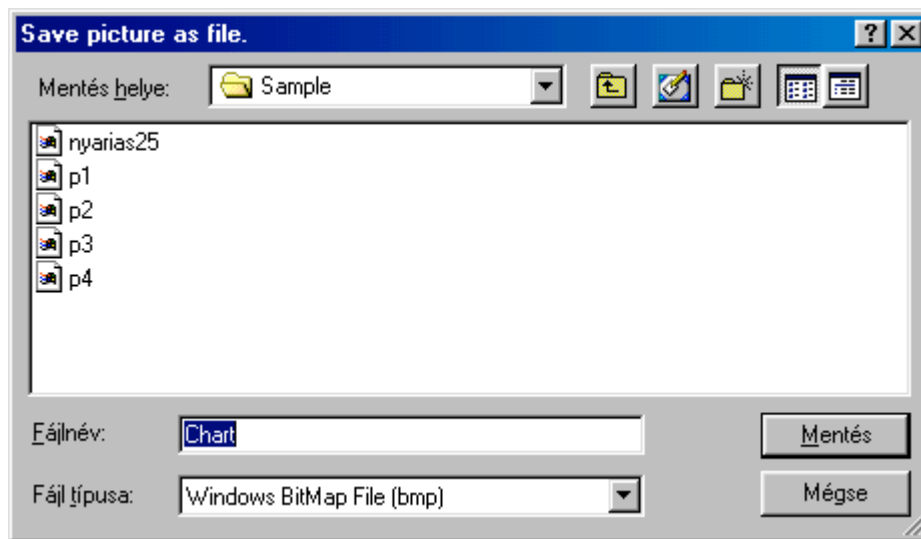
az átlag meghívása, ha még nem aktív

If not(Average1.Checked) Then Average1.Click;

a ciklus lezárása

End;

4.2 MENTÉS



2. Ábra: Az operációs rendszer beépített mentés ablakát használja.

A mentés során a windows operációs rendszer beépített mentés ablakát használtam fel (2. ábra). A program minden mentéskor ajánl egy nevet, amit persze tetszés szerint meg is lehet változtatni. A grafikont és a képet is elmenthetjük Windows Bit Map képformátumban, illetve készíthetünk egy speciális TEXT fájlt, amibe a grafikon adatait úgy tárolhatjuk el, hogy azt egy másik táblázatkezelővel be lehessen olvasni. Ebben az adatpárok egymás alatt külön sorban helyezkednek el: az összetartozó adatpárokat egy szóköz választja el egymástól. Minden görbe adatai előtt egy fejléc is mentésre kerül az azonosítás miatt. Elkerülve a felesleges tárolást és számolást, csak azokat a görbe adatokat menti el a program, amelyek éppen láthatók. Az adatok tízes számrendszerben hatványkitevős alakban vannak tárolva. Például EXCEL programmal is beolvashatóak és szerkeszthetők az adatok. Bővebb információ ezzel kapcsolatban az **Adatmodellek** fejezetben találhatók.

A grafikon képének mentése forráskóddal és magyarázattal:

savedialog ablak címének megadása

```
SaveDialog1.Title:='Save picture as file';
```

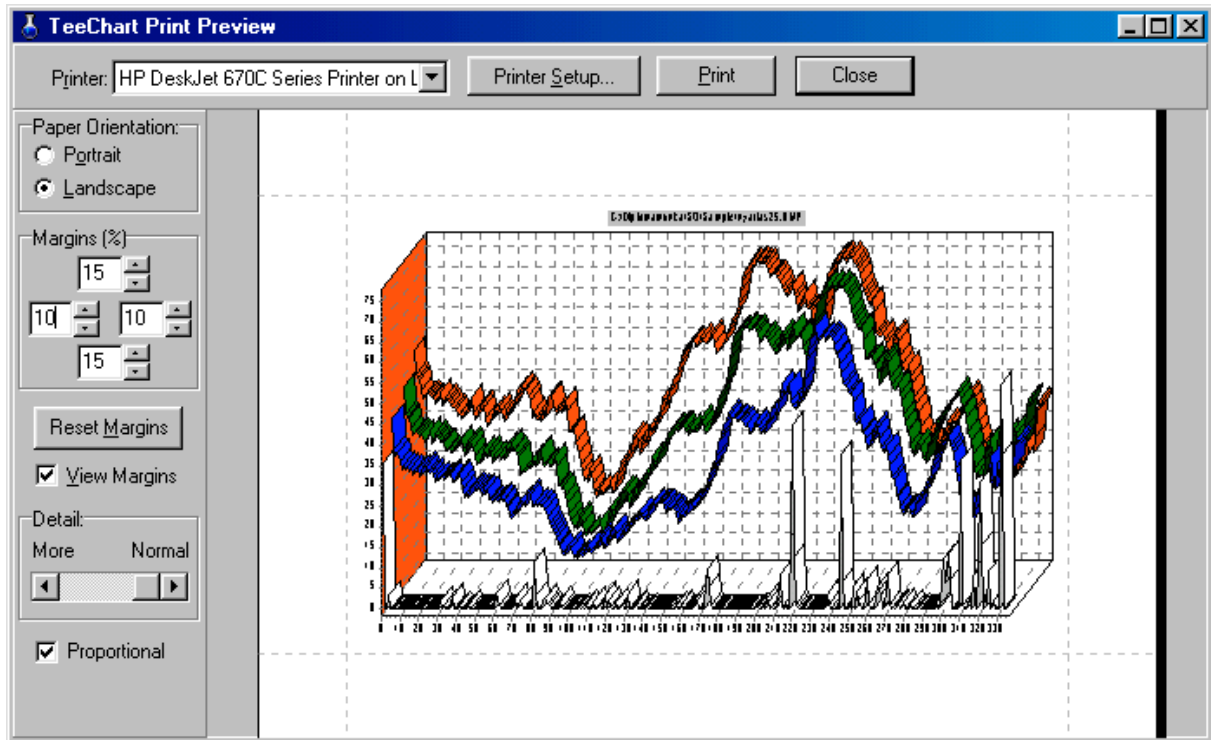
Savedialog ablak OK-val zárva igazat ad vissza

```
If SaveDialog1.Execute then
```

Ha a fenti feltétwl igaz, akkor menti a GRAFIKON objektum képét a savedialog nevével.

```
Chart1.SaveToBitmapFile(SaveDialog1.FileName+'.bmp');
```

4.3 NYOMTATÁS



3. Ábra: A nyomtatás beállításai egy külön ablakban történnek.

A programból a grafikont ki lehet nyomtatni, ha telepítve van nyomtató. A nyomtatást és beállításokat egy speciális grafikonos nyomtatási objektum végzi el, aminek külön ablaka van. Megkapja a GRAFIKON objektumtól a nyomtatási adatokat, és a nyomtatónak megfelelően feldolgozza azt. Be lehet állítani az aktív nyomtatót és ezek tulajdonságait: a lapméretet, a lap állását, a margó vastagságát, a grafikon méretét és a grafikon skáláját.

A grafikon nyomtatása, illetve a beállítások egy ChartPreview objektum segítségével történnek. A programból csak ezt hívjuk meg az alábbiakban két paraméter kíséretében.

az első paraméter a főprogram ablaka, a második a TChart típusú objektum, amit ki szeretnénk nyomtatni

```
ChartPreview(Form1,Chart1);
```


4.4 A KÉP

A programban egy KÉP objektum veszi fel a beolvasott kép pontjait. Minden számolás ettől az objektumtól kéri az adatokat, amik 0 és 255 közötti egész értékek minden fő színek komponensre. Minél kisebb ez a szám, annál sötétebb az adott pont. A pont színét pedig a 3 összetevő egymáshoz viszonyított aránya határozza meg. A KÉP objektum vizuálisan is látható, illetve letiltható. A megkapott képet adott méretűre konvertálja, hogy illeszkedjen a grafikonhoz. Természetesen az adatkommunikáció során mindig az eredeti képmérettel dolgozik a program, hogy ne legyen veszteség.

A kép kimenetként is szolgál, mert van olyan opció, amikor a számolt értékekből állítja elő a képet a program, és ilyenkor ezt a KÉP objektumot használja megjelenítésre. Ezért a látott képet el is lehet menteni fájlba.

A KÉP csak megjelenítésre szolgál, illetve adatok tárolására, ezért a programnak csak néhány tulajdonságát használja fel.

Visible	<i>láthatóság ki-be kapcsolása</i>
Width	<i>a kép szélességének beállítása</i>
Height	<i>a kép magasságának beállítása</i>

4.5 A SZÁMOLÁS

A SZÁMOLÁS objektum kiemelt szerepű, itt történik az adatok feldolgozása. A görbék adatai a nagyobb pontosság érdekében valós számként vannak tárolva. Mivel a különböző görbék kiszámolása sok műveletet igényel, ezeket a program egyszerre végzi el közvetlenül a kép beolvasása után.

Két nagy előnye van ennek a módszernek:

- A számolás során sok azonos művelet is van, így ezeket csak egyszer kell elvégezni (pl. a kép szürkévé konvertálása, átlaggal való számolás).
- A kapcsoló gombokra azonnal megjelenik a kívánt görbe, hiszen már ki van számolva.

A SZÁMOLÁS objektum a kiszámolt értékeket a GRAFIKON objektumnak adja át, és ez tárolja. Ez az objektum egyben a vizuális megjelenítője is a görbéknek.

A számolás objektum forráskódja, és magyarázata egy külön fejezetben szerepel.

4.6 A GRAFIKON

A GRAFIKON objektum szolgál az adatok megjelenítésére. Minden görbe ugyanazon a háttéren jelenik meg. A megkülönböztetés érdekében különböző színnel rajzolja meg a program. A kapcsoló gombokkal a görbék láthatóságát lehet ki- és bekapcsolni, ezzel elérhetjük, hogy bármely görbe teljes méretben csak egyedül legyen megrajzolva. A skálázásnál mindig az összes látható görbe értékei fel vannak tüntetve. Ahhoz, hogy a nagymértékű különbségeket a görbék között kiküszöböljem, ahol csak lehetett %-os ordináta skálát választ a program, így a különböző görbéket is könnyebb összehasonlítani. A legtöbb görbe vonallal van összekötve, kivéve ott, ahol ez értelmezhetetlen; például a hisztogram esetében, amelyen csoportok vannak ábrázolva. Ha a bal egér gombbal kijelölünk egy térrészt, akkor a program a grafikon görbéit felnagyítja.

Az eredeti állapotot úgy érhetjük el, ha a kijelölés negatív irányú, azaz bal felfelé történik. Ha a jobb egér gombot lenyomva tartva a grafikon felett mozgatjuk az egeret, akkor a görbék is vele együtt mozognak.

A grafikon címe mindig a beolvasott és használt képfájl elérési útvonala és neve. Ezért bármikor ellenőrizni lehet a forrás fájlt.

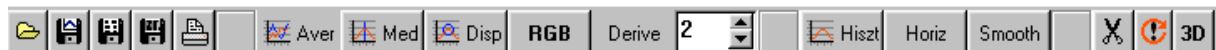
4.7 A MENÜ

A MENÜ objektum egy kapcsolókat és gombokat magába foglaló rendszerezett egység. Fő szerepe, hogy rajta keresztül bármelyik objektumot el tudjuk érni. A főmenükre kattintva almenük jelennek meg. Ezek némelyike ki-be kapcsoló gomb, ezért bekapcsolt állapot esetén egy kis „pipa” jelenik meg mellette. Egyes esetekben még nincs funkciója, vagy értelmetlen a menü gombja, ezért ilyenkor inaktív állapotba kerül, azaz érzéketlen az egérre. Például ha még nem nyitottunk meg semmilyen forrásképet sem, akkor a nyomtatás még értelmetlen. A program ezt ilyenkor nem is engedélyezi. Az almenüben szereplő gombok is csoportosítva lehetnek, ezért ezeket határoló vonalak tagolják.

A menü legkönnyebben egérrel vezérelhető, de a gyorsgombok is aktiválva vannak, mégpedig az aláhúzott betű ALT + betű kombinációval aktiválódik, úgy mintha egérrel kattintottunk volna rá.

A kapcsológombok gyorsabb elérése érdekében eszköztár található a menü alatt.

4.8 AZ ESZKÖZTÁR

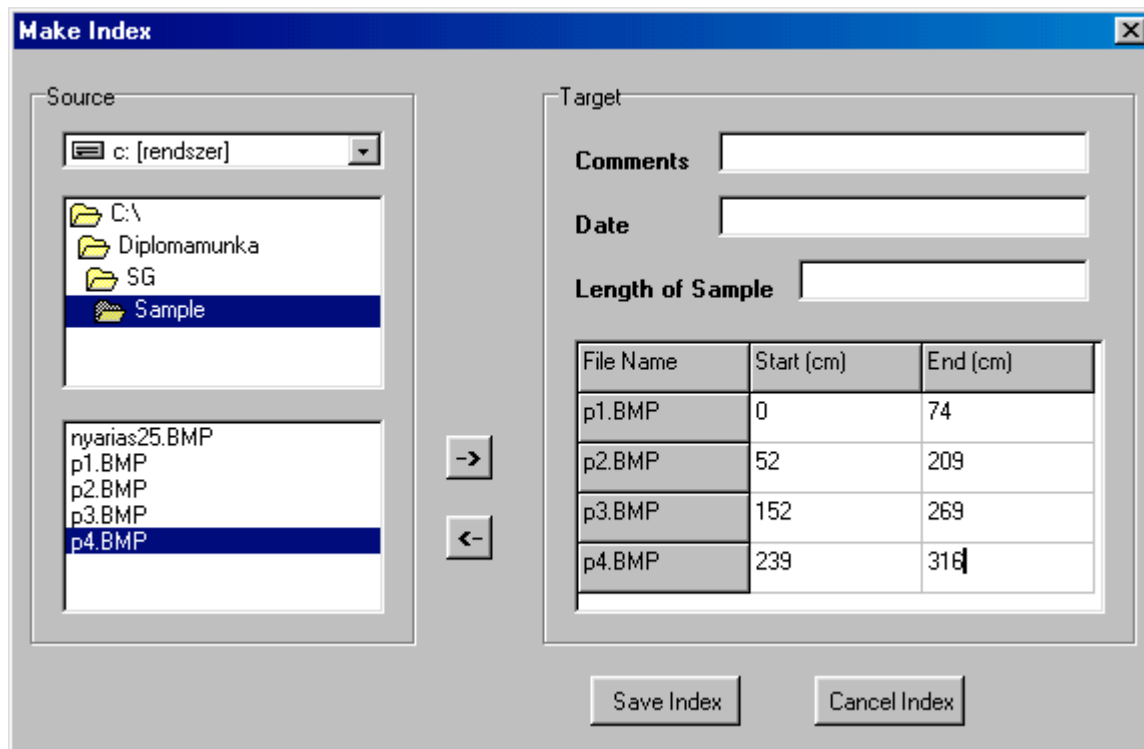


4. Ábra: A programban használt eszköztár.

Az eszköztár legjobban a menüre hasonlít. Itt kapcsolók találhatók logikailag csoportosítva (4. ábra). Némelyikük még képes illusztrációt is tartalmaz, a könnyebb felismerhetőség miatt. A legfontosabb és leggyakrabban használt eszközök kapcsolója található meg itt. Ha egy kapcsoló be van kapcsolva, akkor az megsüllyed, ha pedig nem használható, akkor inaktív, érzéket az egérre. Az eszköztár és a menü állandó kapcsolatban vannak egymással, mivel a legtöbb kapcsoló mind a két helyen megtalálható. Ezért ha az eszköztáron bekapcsolok egy kapcsolót, akkor az a menüben is bekapcsolódik, és akár a menüben is ki lehet kapcsolni azt. A menü tehát hatással van az eszköztárra, de ez fordítva is igaz. Sőt az egész program üzenetek és kapcsolatok hálózata, amelyek mind-mind hatással vannak egymásra.

Az eszköztár a menüben egyben ki- és bekapcsolható; ha valaki nem akarja használni, akkor ne foglalja el a helyet a grafikontól.

4.9 Az INDEX mentése



5. Ábra: Az index fájl készítésére használt űrlap.

A programmal lehetőség van képek automatikus feldolgozására; ilyenkor egymás után beolvassa a képeket és egy grafikonon ábrázolja azok adatait. Mivel itt több adat bevitele is szükséges, ezeket egy külön megnyíló űrlapon kell kitölteni (5. ábra). Itt állíthatók be a mentett képek paraméterei (név, dátum, hossz) is.

Az űrlap két különálló részre tagolt. A bal oldali a forrás, a jobb oldali a cél. A bal oldaliban meg kell keresni a képeket és duplán kattintva, vagy a nyíl segítségével át kell másolni a jobb oldali listadobozba azok neveit. A nevek mellett egy Start és egy Stop mező is található. Ide kell beírni a képek első és utolsó pixelének a pontos távolságát egymáshoz viszonyítva. A hossz mértékegysége nem fontos, de akkor minden kép esetén azonosat használjunk, és az eredményeket is abban kapjuk meg.

Fontos, hogy az elkészített index fájl és a benne felhasznált kép fájlok egyazon könyvtárban legyenek beolvasáskor, különben hibával tér vissza a program.

Az Index fájl speciális szerkezetű, amely egy egyszerű szövegszerkesztővel is elkészíthető. A szintaktikai hiba elkerülése érdekében azonban ajánlatos a SediGraph programmal készíteni el az index fájlt.

Az index fájl szerkezete az **Adatmodellek** fejezetben részletesen tárgyalva van.

A Make index űrlap ideiglenesen készít egy adattáblát, ami 3 oszlopot tartalmaz és annyi sort, ahány képet kiválasztunk. Egy sor tartalmazza a kép nevét útvonal nélkül, Start-ot, és End-et.

Az adattábla meg van jelenítve vizuálisan, így látható a tartalma, és egyben könnyen is szerkeszthető.

A mentés során a tábla tartalmát egy TEXT formátumú fájlba másolja át.


4.10 Az INDEX megnyitása.

Ez annyiban különbözik az eddig tárgyalt megnyitástól, hogy az objektum beolvassa a képeket, és egy üzenetet küld a számolóhoz, ami tudja, hogy ilyenkor még további feldolgozásra váró adatok is jönnek. Az üzenet tartalmazza a kezdeti és az utolsó pont adatait is.

A betöltés után minden képet láthatunk a megjelenítőn, és az adatok feldolgozását nyomon követhetjük.

A számolást tehát ugyanaz az objektum végzi egy kép, illetve indexelt képek esetén is, csak a kapott üzeneteket értelmezve másképp dolgozza fel az adatokat. Ez a megoldás bonyolíthatja ugyan a forráskódot, de a program méretét nagyban lecsökkenti.

4.11 A GYORSSÚGÓ



6. Ábra: A gyorsúgó mindig az aktuális információkat mutatja.

A program egy úgynevezett helyzetérzékeny gyorsúgóval is rendelkezik (6. ábra). Ez a program legalján található. Ha valamely gombra, vagy programrészre állunk az egérrel, akkor a gyorsúgóban megjelenik a neve és rövid leírása. A gyorsúgó valamennyi vizuális komponenssel kapcsolatban van. Az adott komponens érzékeli, ha az egérrel fölötte állunk, és üzenetet küld a gyorsúgónak, egyben pedig elküldi a kiírandó szöveget.

Nagy segítség lehet ez a kezdő programfelhasználóknak.

4.12 Súlyó – Help



7. Ábra: A SediGraph program súlyó rendszerének a tartalomjegyzéke

A program rendelkezik súlyóval. A súlyó 32 bites windows operációs rendszer beépített súlyóját használja ki (WINHLP32.EXE) megjelenítésre, felhasználva annak minden tulajdonságát (7. ábra). A súlyó rendelkezik tartalomjegyzékkel is, amely témakörök alapján vagy pedig kulcsszavakkal könnyíti meg a keresést. Ez a beírt kulcsszót keresi a súlyó oldalakon, és ha többet is talál, akkor felajánlja a választást. Egyes lapokon utalások (linkek) találhatóak, melyek egy másik oldalra mutatnak, amit szintén érdemes elolvasni, mert kiegészítő információt tartalmazhat. Ha az aláhúzott utalásra kattintunk az egérrel, akkor a súlyó automatikusan megnyitja a kapcsolt oldalt [4].

A program Help főmenüjében található CONTENTS (tartalomjegyzék) és a Search for help on... almenü, mely a kulcsszavas keresést aktiválja.

A program help fájjai:

SediGraph.hlp *a sűgó oldalakat tartalmazza Win9x rendszerekre*

SediGraph.cnt *a sűgó tartalomjegyzékét tartalmazza*

Mivel a windows operációs rendszer sűgórendszerét alkalmazza, ez a programtól függetlenül is használható. A SediGraph.hlp-t az operációs rendszerből is elindíthatjuk!

5 A SZÁMOLÁS eljárás kiértékelése

5.1 *Auto Cutting Picture.*

Csak akkor kell a start és stop pontokat megkeresni egy oszlopon belül, ha automatikus vágást végzünk. Ez megvizsgálja az „AutoCut1” gomb állapotát, és ha nincs bekapcsolva, akkor átugorja ezt a részt. A pontok keresése átlag alapján történik, mivel a kép felső és alsó szélének színintenzitásában el kell különülnie a belső résztől. Ezért fentről és alulról is az első olyan pontot keresi a program, amely az átlag közelében van.

Az kapott értékeket két tömb típusú változóban tárolja (Start[i], Stop[i]).

```
If AutoCut1.Checked Then
  for i:=0 to Image1.Picture.Width-1 do begin
    atlag:=0;
    for j:=0 to Image1.Picture.Height-1 do
      atlag:=atlag+(GetRValue(Image1.Canvas.Pixels[i,j])+
        GetGValue(Image1.Canvas.Pixels[i,j])+
        GetBValue(Image1.Canvas.Pixels[i,j]))/3;
    atlag:=atlag/Image1.Picture.Height;
    j:=0;
    Repeat j:=j+1;
      atl:=(GetRValue(Image1.Canvas.Pixels[i,j])+
        GetGValue(Image1.Canvas.Pixels[i,j])+
        GetBValue(Image1.Canvas.Pixels[i,j]))/3;
    Until atl<=atlag;
    start[i]:=j;
    j:=Image1.Picture.Height-1;
    Repeat j:=j-1;
      atl:=(GetRValue(Image1.Canvas.Pixels[i,j])+
        GetGValue(Image1.Canvas.Pixels[i,j])+
        GetBValue(Image1.Canvas.Pixels[i,j]))/3;
    Until atl<=atlag;
    stop[i]:=j;
```



```
end else
for i:=0 to Image1.Picture.Width-1 do begin
    start[i]:=1; Stop[i]:=Image1.Picture.Height-1 end;
```

5.2 Az átlag (számtani közép) kiszámolása.

$$\bar{a} = \frac{a_1 + a_2 + a_3 + \dots + a_n}{n}$$

Az RGB komponensek átlagát veszi, majd függőlegesen átlagolja ezeket. Ezt is eltárolja egy „atlagtomb[i]” változóba azért, mert a későbbiek során még többször szükség lesz az átlagra. A program így gyorsabbá válik, mert nem kell újra kiszámolni ezeket az értékeket, a változókból való kiolvasással.

```
for i:=0 to Image1.Picture.Width-1 do begin
    atlag:=0;
    for j:=Start[i] to Stop[i] do
        atlag:=atlag+(GetRValue(Image1.Canvas.Pixels[i,j])+
            GetGValue(Image1.Canvas.Pixels[i,j])+
            GetBValue(Image1.Canvas.Pixels[i,j]))/3;
    atlagtomb[i]:=atlag/(Stop[i]-Start[i]);
end;
```

5.3 A relatív szórás kiszámolása (RSD%).

$$D = \sqrt{\frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \Lambda + (x_n - \bar{x})^2}{\frac{n}{x}}} * 100$$

Az automatikus vágást figyelembe véve először a függőleges pontok szórását határozza meg, majd megnézi, hogy ez a szórás hány százaléka az átlagnak. Ez azért szükséges, hogy a grafikon különböző értékek esetén is összevethető legyen. A számolt értékeket átadja a **GRAFIKON** objektumnak, s az tárolja. A számolás ideje alatt egy úgynevezett „munka követő doboz” jelenik meg. Az utolsó sor ezt vezérel.

```

for i:=0 to Image1.Picture.Width-1 do begin
  szor:=0;
  for j:=Start[i] to Stop[i] do
    szor:=szor+sqrt((GetRValue(Image1.Canvas.Pixels[i,j])+
      GetGValue(Image1.Canvas.Pixels[i,j])+
      GetBValue(Image1.Canvas.Pixels[i,j]))/3-atlagtomb[i]);
  szor:=szor/(Stop[i]-Start[i]);
  szor:=sqrt(szor); if atlagtomb[i]=0 then atlagtomb[i]:=1;
  szor:=szor/atlagtomb[i]*100;
Series3.AddXY(a+i*dx,szor/2.56,",clPurple);
ProgressBar1.Position:=i;
  end;

```

5.4 A medián érték számolása.

Legyen az alábbi számhalmazunk:

$x_1, x_2, x_3, \dots, x_n$.

$$f(x) = |x_1 - x| + |x_2 - x| + \dots + |x_n - x|$$

Akkor egy számsokaság mediánja az az x szám, amelyre a fenti függvény minimális.

A medián esetén az egy oszlopon belüli pontoknak először az RGB színek átlagát veszi, majd sorba rendezi az adatokat. Ezek közül kiválasztja a számtanilag középső értéket: ez lesz a medián. A program a „buborékos” sorba rendezést alkalmazza. Az adatokat a GRAFIKON objektum tárolja és jeleníti meg.

```

For i:=0 to Image1.Picture.Width-1 do begin
  For j:=0 to Image1.Picture.Height-1 do
    tomb[j]:=(GetRValue(Image1.Canvas.Pixels[i,j])+
              GetGValue(Image1.Canvas.Pixels[i,j])+
              GetBValue(Image1.Canvas.Pixels[i,j]))/3;
  For k:=Start[i] to Stop[i]-1 do
    For l:=Start[i]+1 to Stop[i] do begin
      t1:=tomb[k]; t2:=tomb[l]; tomb[k]:=min(t1,t2); tomb[l]:=max(t1,t2);
    end;
  atl:=tomb[round((Stop[i]+Start[i])/2)];
  Series2.AddXY(a+i*dx,atl/2.56,",clFuchsia);
  ProgressBar1.Position:=i+Image1.Picture.Width;
end;

```

5.5 A hisztogram

A hisztogram esetén egy $tomb[i]$ segédtömböt használ a program, amit először feltölt 0 értékkel. A lehetséges színintenzitásokat felosztja 100 csoportra, és minden színt besorol valamelyikbe. Az adott csoport értéke annyi lesz, ahány oda illő színt talált. A számolás végén a tömbből az adatokat a GRAFIKON objektum kapja meg.

A természetes képek hisztogramja általában Gauss-görbére hasonlít. A görbe maximuma megadja a leggyakrabban használt színárnyalatot.

```

For szín:=0 to 10000 do tomb[szín]:=0;
  For i:=0 to Image1.Picture.Width-1 do begin
    For j:=Start[i] to Stop[i] do begin
      szín:=round((GetRValue(Image1.Canvas.Pixels[i,j])+
        GetGValue(Image1.Canvas.Pixels[i,j])+
        GetBValue(Image1.Canvas.Pixels[i,j]))/3/2.56);
      tomb[szín]:=tomb[szín]+1; end;
    ProgressBar1.Position:=i+Image1.Picture.Width*2;
  end;
For szín:=0 to 100 do
  Series4.AddXY(szín,tomb[szín],"clBlue);

```

5.6 A Horizontal Average

Ez a számolás megegyezik az AVERAGE számolással, csak a képek magassága és hossza van felcserélve. Ha tehát a képet elforgatjuk 90°-al, akkor megkapjuk az átlag (Average) görbét.

```

For i:=0 to Image1.Picture.Height-1 do begin
  ProgressBar1.Position:=i+Image1.Picture.Width*3;
  atlag:=0;
  For j:=0 to Image1.Picture.Width-1 do begin
    atlag:=atlag+round((GetRValue(Image1.Canvas.Pixels[j,i])+

```

```

    GetGValue(Image1.Canvas.Pixels[j,i])+
    GetBValue(Image1.Canvas.Pixels[j,i])/3);
        end;
    atlag:=round(atlag/(Image1.Picture.Width-1));
    Series5.AddXY(i,atlag/2.56,",clLime);
        end;

```

5.7 Az RGB számolás

Itt minden pontnak a három színtkomponensét (vörös, zöld, kék) külön átlagolja függőlegesen, és ezeket külön görbéken jeleníti meg. A görbék színei megegyeznek az ábrázolt összetevők színeivel. Az atlagb, atlagg, atlagr változók az aktuális oszlop színeinek átlagát tárolják. Az automata vágásnak itt is szerepe van.

```

for i:=0 to Image1.Picture.Width-1 do begin
    atlagr:=0; atlagg:=0; atlagb:=0;
    for j:=Start[i] to Stop[i] do begin
        atlagb:=atlagb+GetBValue(Image1.Canvas.Pixels[i,j]);
        atlagg:=atlagg+GetGValue(Image1.Canvas.Pixels[i,j]);
        atlagr:=atlagr+GetRValue(Image1.Canvas.Pixels[i,j]);
            end;
    atlagr:=atlagr/(Stop[i]-Start[i]);
    atlagg:=atlagg/(Stop[i]-Start[i]);
    atlagb:=atlagb/(Stop[i]-Start[i]);
    Series7.AddXY(i,atlagr/2.56,",clRed);
    Series8.AddXY(i,atlagg/2.56,",clGreen);
    Series9.AddXY(i,atlagb/2.56,",clBlue);
    ProgressBar1.Position:=i+Image1.Picture.Width*4;
        end;

```

5.8 A SMOOTH kiszámítása

Egy $2n+1$ pontos simítás képlete az x_i pontban:

$$x_i = \frac{x_{i-n} + x_{i-n+1} + \dots + x_i + \dots + x_{i+n-1} + x_{i+n}}{2n+1}$$

A már kiszámolt AVERAGE adatokat a program egy 5 pontos mozgóátlagos simítással számolja tovább. A simított értékek egy újabb görbén jelennek meg, hogy össze lehessen hasonlítani az eredeti grafikonnal. Az adatokat közvetlenül a GRAFIKON objektum average görbéjétől veszi át és a számolás után egy új görbének az adatmezőit tölti fel.

If Fuz.Flat then begin

i:=0;k:=Series1.MinXValue; l:=0;

Repeat i:=i+1;

d:=0; j:=0;

k:=k+(Series1.MaxXValue-Series1.MinXValue)/(Series1.Count/3);

Repeat j:=j+1; i:=i+1;

d:=d+Series1.YValues.Value[j];

Until (Series1.XValues.Value[i]>k)or(i>Series1.Count-5);

d:=d/j;

Series6.AddXY(k,d,"",clYellow);

l:=l+1;

kep.Width:=l;

end;

Until i>Series1.Count-5;

5.9 A DERIVE kiszámítása

Egy $2n+1$ pontos deriválás képlete:

$$x_i = x_{i-n} - x_{i+n}$$

A negatív értékek, és a jel-zaj viszony javítása miatt negyedik hatványra emelem a derivált értéket:

$$f(x) = x^4$$

*Az egyszerű deriválás nem vezetett célra, mert a görbe meredekségének a statisztikus változása a természetes képek esetén minden pontban közel azonos, illetve a jel-zaj egy nagyságrendbe esik. Ezért a program olyan több pontos deriválást alkalmaz, aminek az intervallumát az **Eszköztárban** lehet állítani egy kapcsolóban. Erre azért volt szükség, mert a kép felbontásától függően az intervallumnak is változnia kellett.*

A jobb jel-zaj viszony érdekében a derivált értékeket a negyedik hatványra emeli, ezáltal a jel sokkal jobban felerősödik, mint a zaj. A páros hatvány miatt a negatív derivált értékek is pozitívvá váltak, így az abszolút érték függvény használata szükségtelenné vált.

Az adatokat az Average görbétől veszi át és egy új derivált görbét hoz létre.

```
for i:=SpinEdit1.Value to Series1.XValues.Count-SpinEdit1.Value-1 do begin
  r:=Series1.YValues.Value[i-SpinEdit1.Value]-
    Series1.YValues.Value[i+SpinEdit1.Value];
  korr:=r*r*r*r/100;
  Series10.AddXY(Series1.XValues.Value[i],korr,",clWhite);
end;
```

6 Az adatmodell

6.1 Előkészületek

Az vizsgálandó objektumokról először digitális képet kell készíteni. Ez manapság már nem nehéz feladat, hiszen a szkennerek, digitális kamerák, webkamerák, digitális fényképezőgépek elterjedőben vannak. A legegyszerűbb módszer, hogy egy hagyományos fényképezőgéppel lefotózzuk a mintát, és az előhívott képet szkennelrel digitalizáljuk, majd nekünk megfelelően megvágjuk. Nagy előnye ennek a módszernek, hogy a fix, nem mozgítható tárgyakat és objektumokat is vizsgálhatjuk vele. Mi ezt, illetve webkamerát alkalmaztunk 640x480 felbontásban és 16 bit színmélységben. 5 cm-es mintahossz már élesen látható volt, vagyis 1 pixel 0,078 mm² ami jónak mondható.

6.2 Bemeneti adatok

A program bemeneti adatként képet fogad. A kép formátuma szabványos Windows Bit Map [7]. A formátumról érdemes tudni, hogy a legelején egy Fej rész található. Ebben a kép tulajdonságai, illetve paramétereit vannak (hossz, szélesség, színek száma). Ezután egy paletta következik. A palettában minden használt szín három fő komponensből épül fel: ezek aránya határozza meg az adott színt. Végül a „színtérkép” következik. Itt mindegyik pontnak meg van határozva a palettán belüli kódja. Ezen formátum előnye, hogy nem alkalmaz tömörítést és elhanyagolást, hiszen nekünk minden egyes pont színe értékes információt hordoz.

A BMP formátum mellett szól még az is, hogy szinte valamennyi windows operációs rendszer alatt futó képszerkesztő program kompatibilis ezzel a szabvánnyal, már csak azért is mert az operációs rendszer felépítő eleme ez a képfomátum. Aki többet szeretne megtudni erről a kép formátumról, az nézze meg az operációs rendszer súgóját.

6.3 Index file

Ez egy speciális file formátum, amit a program alkalmaz. Úgy terveztük meg, hogy a program nélkül is elkészíthető legyen bármely szövegszerkesztővel, ami TEXT formátummal kompatibilis. A file bemeneti és kimeneti adat is egyben.

Azért dolgoztuk ki, hogy a program automatikusan kezelje azt a problémát, amikor több darabban átfedéssel készülnek a képek a talajszelvényről. Ekkor egy Index.ndx file-t lehet készíteni a programmal. A kimeneti és bemeneti filenévénél nem kötelező meghagyni a program által ajánlott nevet.

A formátum szintaxisa: először egy FEJ információt készít, névvel, dátummal, szelvény hosszával, képek számával, ezután a TÖRZS következik, ahol a kép neve található kiterjesztéssel (bmp), majd új sorokban az első pont, és az utolsó pont. Ez annyiszor ismétlődik, ahány kép van.

Itt látható egy Index file, mellette zárójelben a magyarázattal:

[Header Information]	<i>(fej azonosító)</i>
Backwater:	
Röszkei-Holt-Tisza	<i>(azonosító név)</i>
Date:	
2001.01.10	<i>(a mintavételezés dátuma)</i>
Length of Sample:	
200-250	<i>(a teljes hossz: ezek az értékek cm-ben értendők)</i>
File Number:	
4	<i>(ennyi képből áll össze a szelvény)</i>
[File Index]	<i>(file azonosító)</i>
0-18.bmp	<i>(az első kép neve kiterjesztéssel)</i>
200	<i>(az első kép baloldali első pontjának pontos értéke)</i>
218	<i>(az első kép jobboldali utolsó pontjának pontos értéke)</i>
12-29.bmp	<i>(a második kép neve kiterjesztéssel)</i>
212	...
229	
26-40.bmp	
226	
240	
32-50.bmp	
232	
250	<i>(az utolsó kép utolsó, jobboldali pontjának pontos értéke)</i>

6.4 Kimeneti adatok

A program által készített grafikonokat és a számolás alapján előállított képet el lehet menteni, amely ugyancsak BMP formátumú, mint a bemenetül szolgáló kép.

6.5 Adat file

A program a kiszámolt értékeket szükség szerint nemcsak ábrázolni tudja, hanem akár el is menti egy kimeneti fájlba DAT kiterjesztéssel. A mentett file beolvasható a legtöbb táblázatkezelővel (pl. EXCEL), így utána tetszőlegesen felhasználható.

Minden adatblokknak először a fejléce látható, benne az x koordináta, és a megnevezés. Utána soronként az összetartozó adatpárok SZÓKÖZZEL ELVÁLASZTVA. Az adatok tízes számrendszerben hatványkitevős alakban vannak. Ha egy ábrázolási mód nincs bekapcsolva a programban, akkor a program annak csak a fejlécét menti el, így a számunkra lényegtelen adatok nem növelik meg feleslegesen a file méretét [5].

Lássunk egy adat file-t magyarázattal:

This_Is_SediGraph_Data_File	(file fejléce)
X-Chordinata Average	(átlag fejléce)
0.00000000000000E+0000 3.04687500000000E+0001	(az első pont)
9.93589743589744E-0001 3.26923076923077E+0001	(a második pont)
...	
1.54006410256410E+0002 4.70552884615385E+0001	(az utolsó pont)
X-Chordinata Median	(medián fejléce)
X-Chordinata Dispersion	(szórás fejléce)
0.00000000000000E+0000 5.11474045574542E+0000	
9.93589743589744E-0001 4.07546567607148E+0000	
...	
1.54006410256410E+0002 2.99399848874219E+0000	
X-Chordinata Red	(vörös szín fejléce)
X-Chordinata Green	(zöld szín fejléce)
X-Chordinata Blue	(kék szín fejléce)

X-Chordinata Smooth

(simítás fejléce)

2.96166173570020E+0000 3.26171875000000E+0001

5.92332347140039E+0000 3.28125000000000E+0001

...

1.51044748520710E+0002 4.30288461538462E+0001

X-Chordinata Average_Derivate

(deriválás fejléce)

0.00000000000000E+0000 2.56000000000000E+0001

9.93589743589744E-0001 2.02514792899408E+0001

...

1.44070512820513E+0002 9.69467455621303E+0000

X-Chordinata Horisontal

(horizontális fejléce)

0.00000000000000E+0000 3.98437500000000E+0001

1.00000000000000E+0000 3.94531250000000E+0001

...

1.40000000000000E+0001 4.21875000000000E+0001

X-Chordinata Histogram

(hisztogram fejléce)

0.00000000000000E+0000 0.00000000000000E+0000

1.00000000000000E+0000 0.00000000000000E+0000

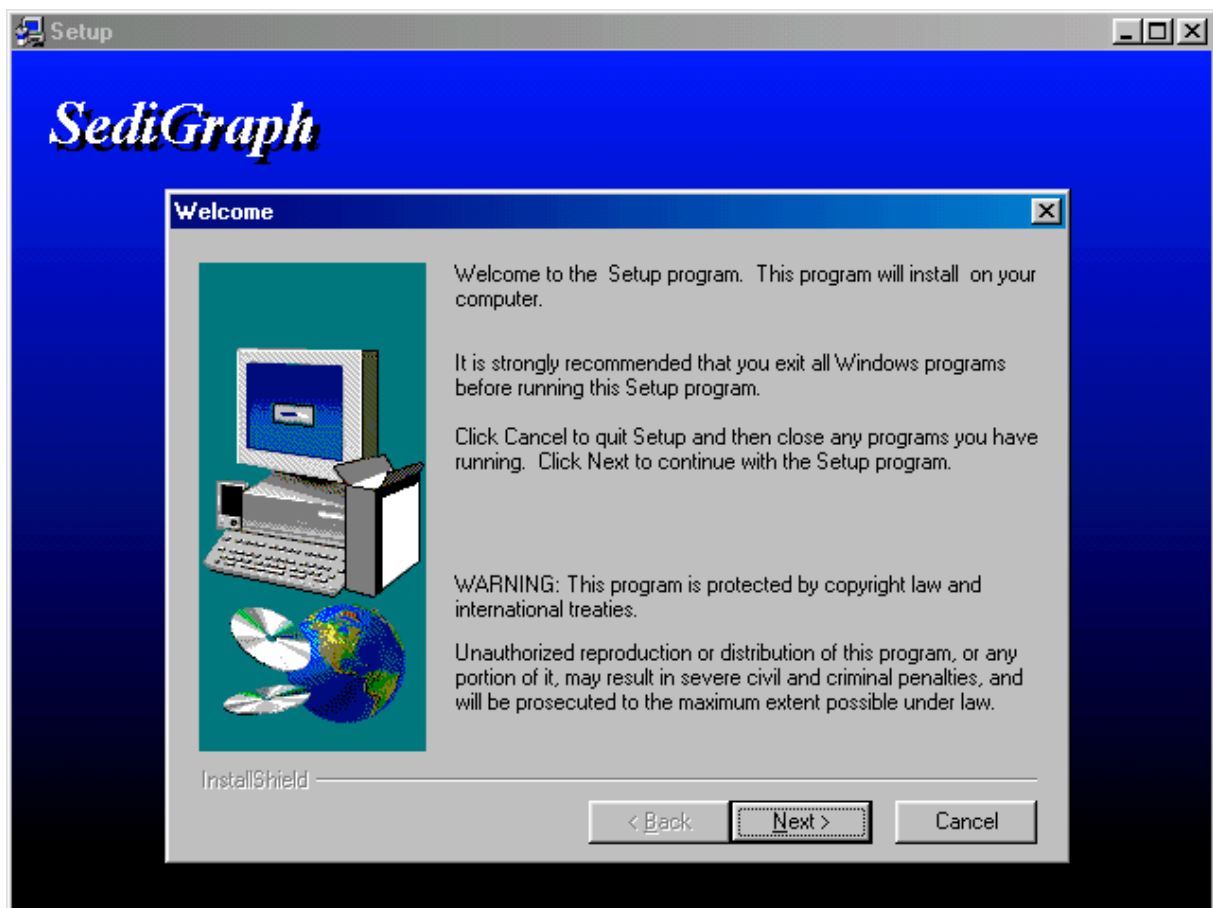
...

1.00000000000000E+0002 0.00000000000000E+0000

7 A program ismertetése felhasználói szemmel

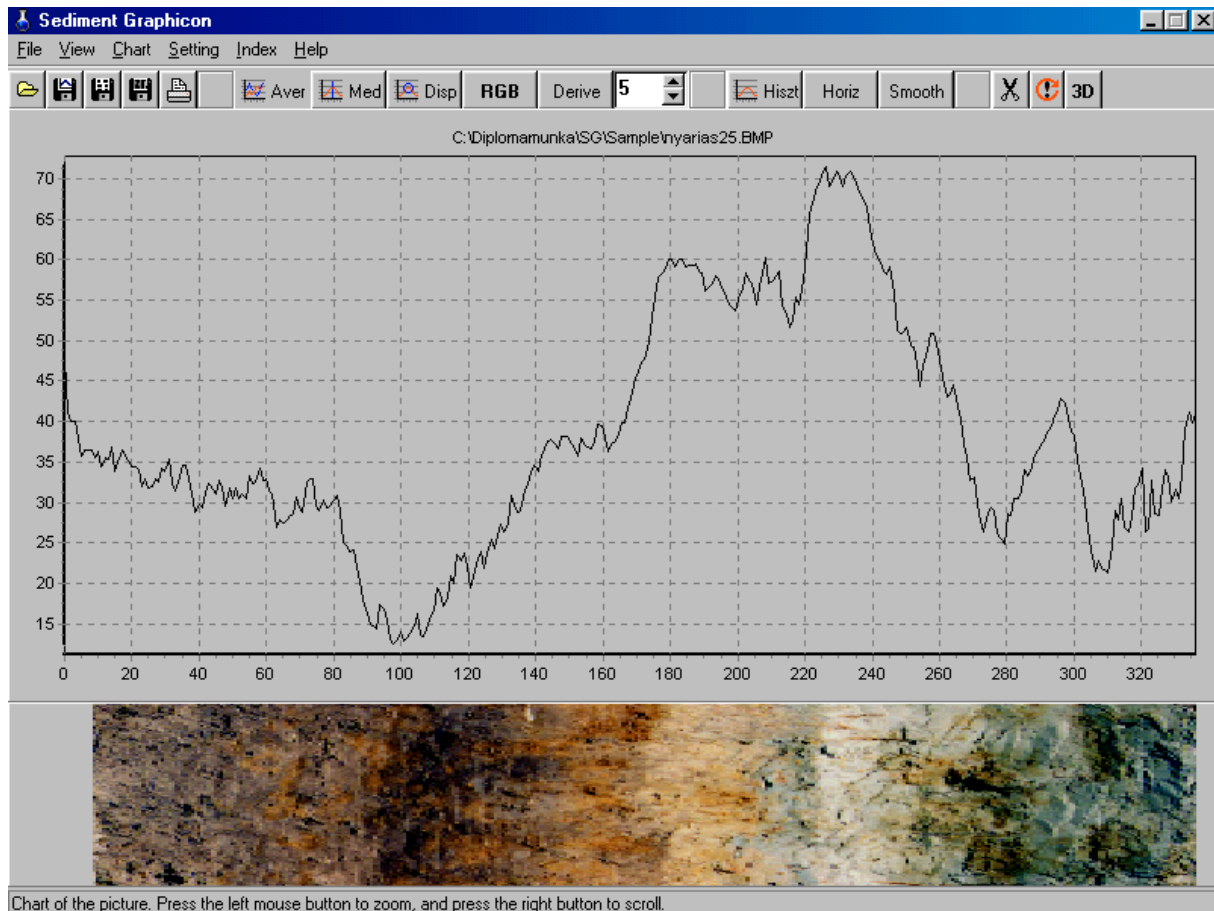
7.1 A SediGraph program telepítése

1. Először vegye ki a CD lemezt a dobozából, és tegye be a CD-rom meghajtóba!
2. Indítsa el a CD lemezről a Setup.exe programot! Kattintson a Sajátgép ikonra, majd CD meghajtó, és végül Setup.exe!
3. Kövesse a telepítő utasításait (8. ábra)! Általános telepítésnél fogadjunk el mindent, amit felajánl a telepítő.
4. Telepítés után a Start menü Programok között találhatjuk a SediGraph programot. Indítsuk el!



8. Ábra: A telepítés során megjelenő ablak.

7.2 A program felépítése



9. Ábra: A Sedigraph program főablaka

A programnak öt fő látható része van (9. ábra):

- menü
- eszköztár
- grafikon
- kép
- gyorsítógépek

7.2.1 Menü

A menü arra szolgál, hogy a program egyes részeit elérjük és parancsokat adhassunk. A menü gombjainak felirata információt ad, hogy mire is való. A menü réteges felépítésű, mindig láthatóak a fő részek, amelyek csoportokba foglalják az almenü gombjait. A logikai felépítés az almenükben is benne van, ezért határoló vonalak találhatóak benne.

A menüt egérrel a legkönnyebb kezelni, csak rá kell kattintani a bal egérgombbal a kiválasztott részre. A főmenüből nyílnak az almenük. Minden menügomb feliratán aláhúzott betű található. Az ALT + „adott billentyű” egyidejű lenyomása helyettesíti az egeret. Az egyes menügombok kapcsolók is egyben, vagyis a ki és bekapcsolt állapotot is megjelenítik. A bekapcsolt állapotot egy kis „pipa” jelzi a.

Néha a menü egyes részeinek nincs szerepe. (Ilyen például a mentés, amikor még nem olvastunk be semmit.) Ekkor ezek a menü gombok „inaktívak”, érzéketlenek az egér és billentyű hatására, és csak halványan jelennek meg.

7.2.2 Az eszköztár (ToolBaar)

Az eszköztár szerepe a programban, a gyorsabb programhasználat. A leggyakrabban használt gombok és kapcsolók találhatóak itt. Gyorsaságát annak köszönheti, hogy egyszeri kattintással a menüben lévő pontokat működteti. A gombok feliratán kívül némelyiken kis ábra is látható, amely ugyancsak a gyorsabb felismerést segíti elő.

Ezek is logikai csoportokba vannak sorolva. Bekapcsolt állapotban lesüllyed a gomb. Ha nincs tartalmi jelentősége, akkor sötét színű lesz, és érzéketlen az egérre. Alapvető különbség a menü és eszköztár gombjai között, hogy az utóbbit csak egérrel vezérelhetjük.

7.2.3 A grafikon (Chart)

A programnak az egyik legfontosabb része. Ezen az ablakon keresztül grafikusán láthatjuk a vizsgált képből előállított értékeket [6]. A grafikonok egyazon koordináta-rendszerben vannak megjelenítve. Csak a bekapcsolt görbék láthatóak egy időben, s hogy jobban elkülönüljenek, más-más színnel szerepelnek.

A grafikont egérrel kétféleképpen lehet kezelni:

Nagyítás: A bal egér gombot lenyomjuk, és lenyomva tartva a kinagyítandó részletet kijelöljük bal fentről a jobb lenti irányban. Ha az eredeti méretet szeretnénk visszakapni, akkor a kijelölés negatív irányú legyen, vagyis jobb lentől bal fenti irányú, természetesen a bal egér gombbal.

Mozgatás: A jobb egér gombot nyomva tartva mozgassuk az egeret a megfelelő irányba. Ekkor a görbe együtt mozog az egérrel.

Ha egy görbét elemezni szeretnénk, akkor érdemes az összes többit kikapcsolni, hogy ne zavarjon.

7.2.4 A kép (Picture)

A vizsgálni kívánt kép látható itt. A program minden beolvasott képet átméretez a megjelenítéshez, ezért a görbék pontjai az alatta elhelyezkedő képpontokhoz tartoznak. Ez nagy segítség a kiértékelésnél, és szemmel is összevethető a kapott eredmény.

7.2.5 A gyorsúgó

A program ezen része a gyors segítségnyújtást szolgálja. Ha a program bármely része fölé állunk az egérrel, akkor itt információ jelenik meg róla. Az információ rövid, lényegretörő szöveges üzenet. A menük és eszköztárak gombjai is mind szerepelnek a gyorsúgóban, megkönnyítve ezzel a munkát.

A program egy összetett súgórendszerrel is rendelkezik, ahol részletesebben is tárgyalva van minden rész, illetve kulcsszó alapján is kereshetünk benne. Ezt a menü Help részénél lehet megtalálni.

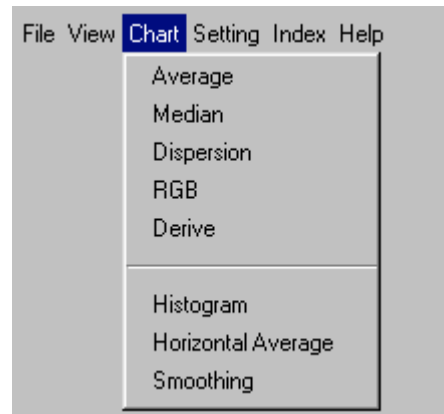
Sajnos a gyorsúgó nyelve angol, de a rendes súgó már magyar nyelvű.

8 Menürendszerek tárgyalása felhasználóknak

8.1 A menüpontok

A program rendelkezik menüvel. Minden utasítás és opció elérhető a menükön keresztül. A menüpontok legegyszerűbben egérrel kezelhetők. A főmenükre kattintva almenü jelenik meg, ezek közül választhatunk.

A programban a kezelhetőség gyorsítása miatt Eszköztár (ToolBar) is található. Ez a menü alatt helyezkedik el, és a legtöbbet használt menük gombjai találhatóak itt ikonok formájában.



10. Ábra: A Sedigraph menürendszere.

8.2 File

Itt a bemeneti és kimeneti adatforgalommal kapcsolatos almenük találhatóak.

8.2.1 Open

A számoláshoz szükséges Windows Bit Map (bmp) formátumú kép fájlt itt kell hozzáadni a programhoz. A képen a vizsgálandó mintákat vízszintesen kell elhelyezni. A kép mérete nincs fixálva, de túl nagy képek esetén a program lelassulhat; ekkor tanácsos egy képszerkesztő programmal csökkenteni méretét. Ez általában nem okoz lényeges eltéréseket, csak kevesebb pontból állnak a grafikonok.

A kép színkezelése nagyon fontos, mert ebből nyerjük ki az információt. Legalább 256 színt használjunk, de erősen ajánlott a 16 bites színkezelés. Ha a színek nem túl dominánsak, ajánlatos szürke árnyalatú képet készíteni. A színes képnél a program külön tudja ábrázolni a három főkomponenst, (RGB) a vörös, a zöld és a kék színeket.

8.2.2 Save Chart

Itt a grafikont lehet elmenteni képformátumban (bmp). Az elmentett kép mérete megegyezik a látott kép méretével pixelben, így aki nagyobb felbontást használ, nagyobb és jobb felbontású képet nyer. Dolgozatok, mérési naplók kiegészítéseként szolgálhat nagy segítségre az így készített kép.

Tipp: Ha nagyobb grafikont szeretne, kapcsolja ki a View menüben található gombokat.

8.2.3 Save Data

Adatfájlt készíthetünk. A fájl TEXT formátumú, az adatpárokat egymás után veszi szóközzel elválasztva. Csak azok a grafikonok jelennek meg a fájlban, amelyeket bekapcsolunk, hogy helyet és időt takarítsunk meg. A program automatikusan DAT kiterjesztést ad, de ez tetszőlegesen felülbírállható.

Tipp: Excel használható az adatok további feldolgozásához. Beolvasáskor szóközzel elválasztott adatokat kell beállítani!

8.2.4 Save Picture

A program a képekből nyert információkból egy újabb képet állít elő. Ez egy vonalas szerkezetű „spectrumos” kép, amely a simítás (Smooth) menüpontra aktiválódik, és a beolvasott kép helyén jelenik meg. Ezt az előállított képet lehet elmenteni itt (bmp) képformátumban.

Tipp: A mentés előtt nem kell bekapcsolni a simítás (Smooth) menüpontot, mert a program figyeli ezt és magától megteszi.

8.2.5 Print

A print kapcsolóra kattintva a grafikont nyomtathatjuk ki, ha a számítógéphez van telepítve nyomtató. Először egy segédablak jelentkezik be, ahol a nyomtatás tulajdonságait lehet beállítani, miközben láthatjuk a nyomtatandó lapot is.

Beállíthatók:

- az aktív nyomtató
- a nyomtató beállításai

- a papír orientációja
- a margók %-osan
- a grafikon beosztása, finomsága

Tipp: A kinyomtatandó lap látható, és sok beállítást magán a képen is el lehet végezni: többek között az igazítást és a nagyítást.

8.2.6 Exit

Bezárja az ablakokat, és kilép a programból. Figyelem! Az el nem mentett adatok elvesznek, ezért minden kilépés előtt győződjön meg arról, hogy munkája el van mentve.

Tipp: Gyorsabban ki lehet lépni a programból, ha az egérrel a jobb felső sarokban lévő x-re kattint.

8.3 View

Itt a programban található egységek láthatóságán állíthatunk.

8.3.1 ToolBaar

Ez a kapcsoló gomb az Eszköztárat kapcsolja be és ki. Az eszköztárban lévő valamennyi gyorsgomb megtalálható a menükben csoportosítva; csak a gyorsabb és egyszerűbb kezelhetőség miatt lett létrehozva.

Indításkor a program bekapcsolja.

Tipp: Ha kikapcsoljuk az eszköztárat (ToolBaar), megnő a grafikon területe.

8.3.2 Picture

Ezzel a kapcsolóval a vizsgálandó képet lehet be- és kikapcsolni. A képet össze lehet hasonlítani a kapott grafikonokkal. A beolvasott képet nagyítja, vagy kicsinyíti a program a fix képernyőméret és grafikon miatt.

Indításkor a program bekapcsolja a láthatóságot.

Tipp: Ha kikapcsoljuk a képet (Picture), megnő a grafikon területe.

8.4 Chart

Itt a grafikonnal kapcsolatos kapcsolók találhatóak.

8.4.1 Average

Átlag. A képet először szürkévé konvertálja, majd függőlegesen átlagolja a pontokat, és százalékos skálává alakítja úgy, hogy 0 a teljesen fekete szín, és 100 a teljesen fehér. Ha a képben vízszintes irányban eltérés mutatkozik, akkor ott a görbe meredeksége megváltozik. Ha egy görbe csak 1-2%-ban változik, akkor a kép homogén.

Tipp: A szórás (Dispersion) felvilágosítást adhat, hogy a kép függőlegesen mennyire homogén.

8.4.2 Median

A képet először szürkévé konvertálja, majd a függőleges pontok mediánját veszi. Az ordináta tengely beosztása 0 és 100 közé esik, mivel a teljesen fehér színt 0-nak, a teljesen feketét 100-nak kalibrálja a program. Ha az átlagtól (Average) eltérés mutatkozik, akkor a kép függőleges irányban nem homogén.

Tipp: Az átlaggal való összehasonlítást mindig érdemes megnézni.

8.4.3 Dispersion

Szórás. A képet először szürkévé konvertálja, majd a függőleges pontok szórás százalékát adja vissza. Minél nagyobb ez az érték, relatíve annál nagyobb a pontok szórása függőleges irányban.

Tipp: Ha például egy talajszelvény közepén gyökér található, akkor hamis eredményt ad a gyökér más színe miatt. Igen ám, de a szórás itt nagymértékben meg fog nőni. Ezért mindig érdemes megvizsgálni a minták szórását, és összevetni a kapott értékkel.

8.4.4 RGB

Vörös, Zöld, Kék. Színes képek esetén, amikor jól láthatóak az eltérések, lehetőség van a három fő színösszetevő (vörös, zöld, kék) alapján is vizsgálni a mintát. Ekkor a képnek csak az adott színt komponense van függőlegesen átlagolva. A kapcsolóval mind a

három fő szint egyszerre lehet megjeleníteni és eltüntetni. A grafikon rajzolásánál értelemszerűen olyan színnel vannak megrajzolva, mint amit jelképeznek. Itt is %-ra van konvertálva az ordináta tengely.

Tipp: Általában sokkal több információt hordoz a különböző színek aránya, mint az értékük nagysága.

8.4.5 Derive

Derivált. A struktúrák legjobban a határvonalakkal jellemezhetők. Ezek az átlagból (Average) származtathatók deriválással. A deriválás speciális kell legyen, mert az átmenet sokszor statisztikus, nem éles. Ezért több pontos deriválás történik, amit állítani lehet az eszköztárban (ToolBaar). A jel-zaj viszony javítása érdekében exponenciális erősítést alkalmazok.

Tipp: Különböző képek esetén más és más beállítás kell a deriválásnál, hogy élesebb jelet kapjunk. Ezért mindig érdemes megvizsgálni a beállítást.

8.4.6 Hisztogram

A képet először szürkévé konvertálja, majd végignézi minden színintenzitást, megszámlolja, hogy az hányszor fordul elő. Így itt minden egyes pixelt külön megszámlol. Ez általában egy Gauss görbét ír le. A legnagyobb értékű szín fordul elő a legtöbbször a képen.

Tipp: A hisztogram felvilágosítást ad, hogy a képet helyesen készítettük-e el; a Gauss görbének ugyanis középre kell esnie. Ha nem így van, akkor a kép túl világos, vagy túl sötét.

8.4.7 Horizontal Average

Vízszintes átlag. Itt a program a szürkévé konvertált képet vízszintesen átlagolja. Ez a függőleges eltérésekről ad felvilágosítást.

Tipp: A Horizontal Average és Average ugyanazt az eredményt adja, ha a képet 90 fokkal elforgatjuk a két mérés közben.

8.4.8 Smoothing

Simítás. A kapcsoló hatására az átlag (Average) görbét 5 pontos mozgó átlaggal simítja. Az új értékek alapján rekonstruálja a bemeneti képet, és egy vonalas szerkezetűt (spektrumos) állít elő. Az új kép esetén a színekülönbségeket széthúzza a fehértől a feketéig, így a kisebb eltérések is jobban láthatóvá válnak.

Tipp: Az így előállított képet el lehet menteni a File menü Save Picture almenüjével.

8.5 Setting

Itt néhány program beállítás található.

8.5.1 Auto Cutting Picture

Automatikus képvágás. Ha a képet nem vágjuk meg kézzel a kiértékelés előtt, akkor lehetőség nyílik automatikus vágásra, melyet a program végez el egy vágó algoritmus alapján. Ennek feltételei a következők:

- A kép hosszában középen helyezkedjen el a talajszelvény.
- A kép két széle fehér illetve világosabb legyen, mint a talajszelvény.

Ezek alapján a program az adott függőleges oszlopban megkeresi a felső és alsó vágási pontokat, és csak a kettő közötti pixelelkekel dolgozik.

A kapcsoló bekapcsolása után újra be kell olvasni a forrásképeket, hogy a vágás működjön. Kikapcsolásig az összes képet vágni fogja.

Tipp: Ha a minta elé egy fehér kitakarású ablakot helyezünk, és azt digitalizáljuk be, akkor ezzel a kapcsolóval megspórolhatjuk a kézi vágást, ami ráadásul szubjektív hibát is eredményezhet.

8.5.2 Horizontally Reverse

Vízszintes megfordítás. Lehetőség van a grafikonok vízszintes tükrözésére is. Ez akkor lehet hasznos, ha a képen nem a sötétebb foltokat, hanem a világosabb részeket vizsgáljuk.

Tipp: Megfordításnál a pontok ordináta értékei nem változnak, csak a tengely két vége cserélődik fel.

8.5.3 3 Dimension Chart

3D grafikon. Ha a grafikonon egyszerre sok görbe van feltüntetve, és ezek nem nagyon térnek el, akkor elfedhetik egymást. Ezen kapcsoló bekapcsolásával a grafikon egyes görbéi eltolva rajzolódnak meg, így jól láthatóvá válnak.

Tipp: A grafikon így jobban mutat, színesebb, és bemutatók illusztrálására jobban fel lehet használni. Mindemellett a File menü Save Chart almenüjével el lehet menteni kép formátumban.

8.6 Index

Indexeléssel kapcsolatos almenük található itt.

8.6.1 Make Index

Index fájl készítése. A program lehetőséget ad arra, hogy átfedésben lévő talajszelvényekhez „Index” fájlt készítsünk, amivel a képeket összefűzhetjük, és egyszerre ábrázolhatjuk. Az index készítése egy külön úrlapon történik. A fájl neve automatikusan Index.ndx, de ezt bármilyen tetszőleges névre átírhatjuk.

FONTOS: Az index fájlnak mindig abban a könyvtárban kell lennie, ahol a forrás képek találhatóak!

Tipp: Az index fájl speciális szerkezetű, de bármely egyszerűbb szövegszerkesztővel el lehet készíteni. A leggyorsabb azonban a programmal létrehozni, hiszen így a szintaktikai hibákat elkerülhetjük.

8.6.2 Open Index

Index fájl megnyitása. Az elkészített index fájlt itt lehet beolvastatni a programmal és feldolgoztatni a benne felsorolt képeket. Ez hosszabb időt is igényelhet a képek méretétől

függően. Ha automata vágást (auto cutting picture) szeretnénk, akkor még a beolvasás előtt kapcsoljuk be ezt a kapcsolót!

Tipp: A beolvasott index fájlnak abban a könyvtárban kell lennie, ahol a hivatkozott képek találhatóak. Ezért ha beolvasás során nem találja a képeket, manuálisan vizsgáljuk meg, hogy az előbbi feltételnek eleget tesz-e.

8.7 Help

Itt a felhasználói segítség található.

8.7.1 Contents

Tartalomjegyzék. A sűgó tartalomjegyzéke ez, amely csoportokba foglalja a program részeit. Itt katalógus szerint kereshetünk.

8.7.2 Search for help on...

Keresés a sűgóban. Itt címszavak, illetve kulcsszavak alapján lehet keresni. Ha egy kulcsszó több részben is megtalálható, akkor a sűgó mindet megjeleníti, hogy válasszunk belőle.

Tipp: Ha egy adott sűgóban kiemelt és aláhúzott szót találunk, akkor az hivatkozás. Ha erre rákattintunk, akkor a kijelölt címszóval kapcsolatos információ jelenik meg.

8.7.3 About...

Itt a programról található néhány információ:

- a program teljes neve
- a program verziószáma
- a készítés ideje
- a készítő neve
- a készítő elérhetősége

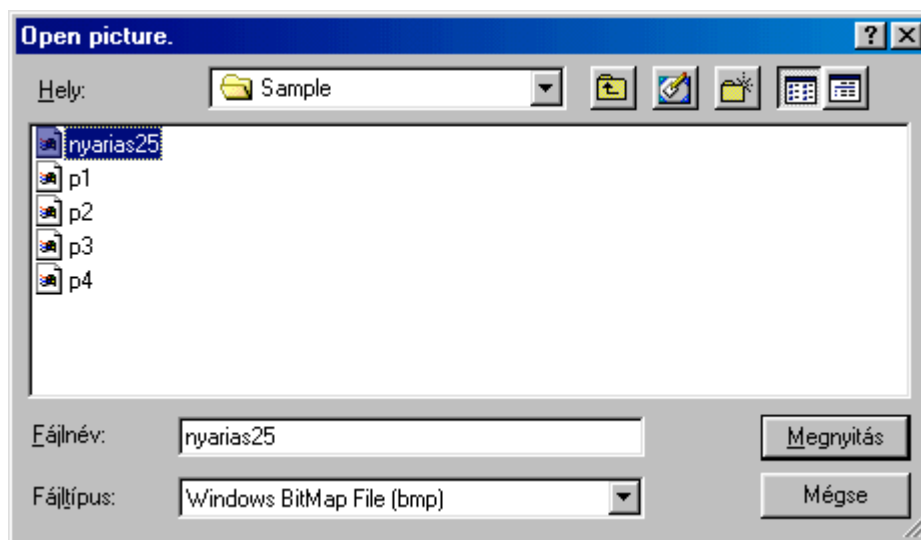
Tipp: Ha a program megnyerte tetszését, érdemes felvenni a készítővel a kapcsolatot, hogy készült-e nagyobb tudású újabb verzió.

9 A SediGraph program ismertetése példán keresztül

A program több példaképet is telepít a Sample (Minta) Könyvtárba. Először a nyarias25.bmp kép legyen a vizsgálandó mintánk.

Indítsuk el a programot a Start menü Programok, SediGraph-ra kattintva! A program egyből teljes méretben jelentkezik be. A grafikon és a kép helye ekkor még üres. A kép beolvasása kétféleképpen is lehetséges: vagy a File menü Open, vagy az eszköztárban az ezzel egyenrangú gombbal. Jelen esetünkben ez a bal szélső. Ne feledjük, mindig segítségünkre van a gyorsúgó.

Egy Open picture ablak jelenik meg, és kéri a kép nevét (11. ábra). Adjuk meg az útvonalat és a kép nevét, majd OK.



11. Ábra: A megnyitás ablak képe.

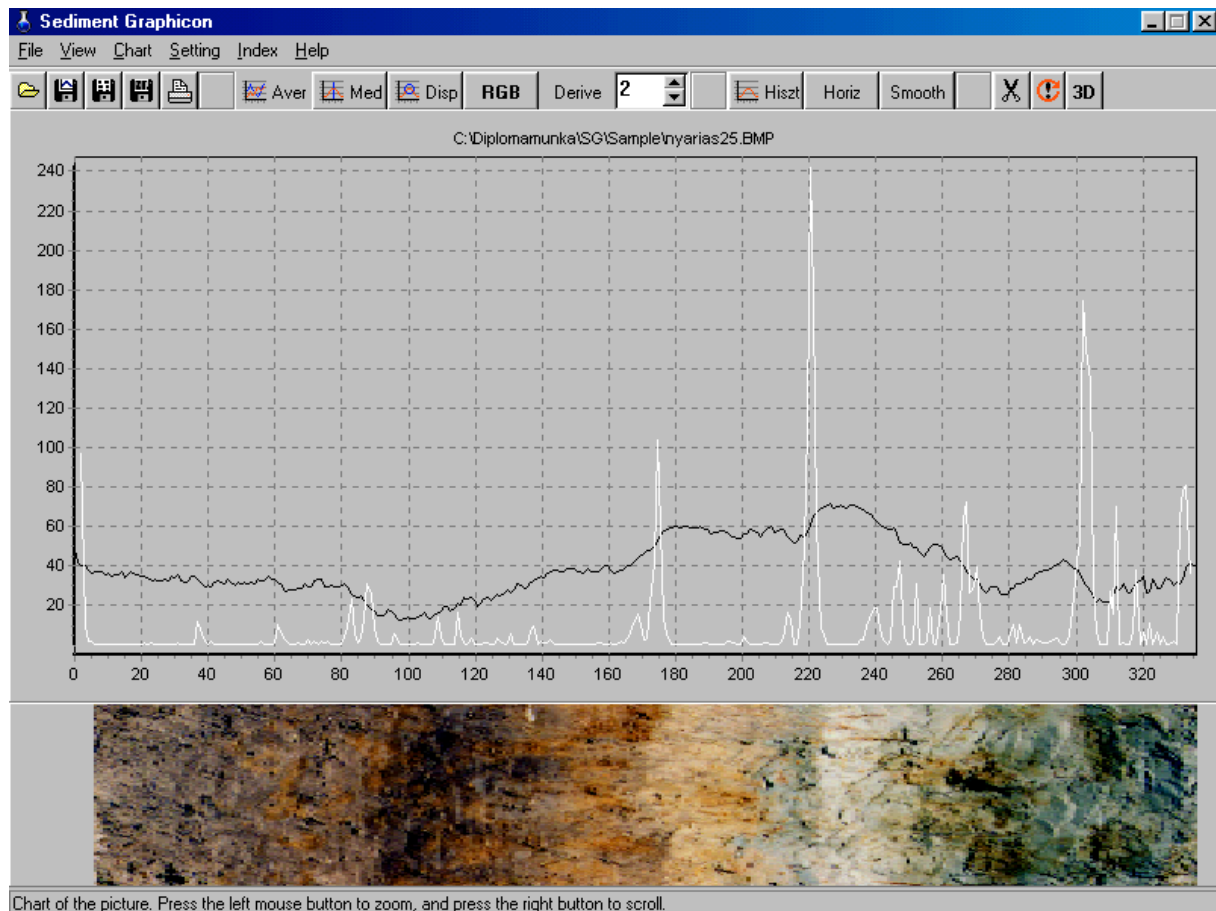
Ha jól dolgoztunk, akkor a program alján megjelenik a minta képe, majd számolni kezd, amit egy mozgó kék sáv jelez. A program akkor van kész, ha végig ér a kék sáv.

A program ezzel a fő számolásokat és kiértékeléseket elvégezte, s az átlag kapcsolót automatikusan bekapcsolva a grafikonon megjelenik egy görbe (12. ábra). Ez a görbe minél magasabb értékű az adott pontban, annál világosabb abban az oszlopban a kép átlag színintenzitása. Mindig a szerkezetek közötti határvonalat keressük, ezért a görbe gyors változása fontos a számunkra.



12. Ábra: A program nézeti képe az átlag (average) görbével.

A Derive (derivált) kapcsoló bekapcsolásával egy fehér görbe is megjelenik a grafikonon (13. ábra). Ezen a görbén csúcsok találhatók. A csúcs helyei a színek változásának határán vannak, a magasságuk pedig arányos az átmenettel; vagyis minél nagyobb egy Derive görbe csúcsmagassága, annál gyorsabb az átmenet a különböző szerkezetek között. A derive kapcsoló mellett az eszköztárban található egy léptető ablak. Benne egy egész számmal, amelyet ha változtatunk, akkor az a deriválás módját befolyásolja. Próbáljunk meg állítani a kapcsolón az egérrel, és figyeljük meg közben a Derive (derivált) görbe alakját is. Minden mintának a képe más és más a természetes jellege miatt, de ezzel a beállító gombbal úgynevezett „finomhangolással” élessé lehet tenni az átmeneteket.



13. Ábra: Az átlag és a derivált görbe.

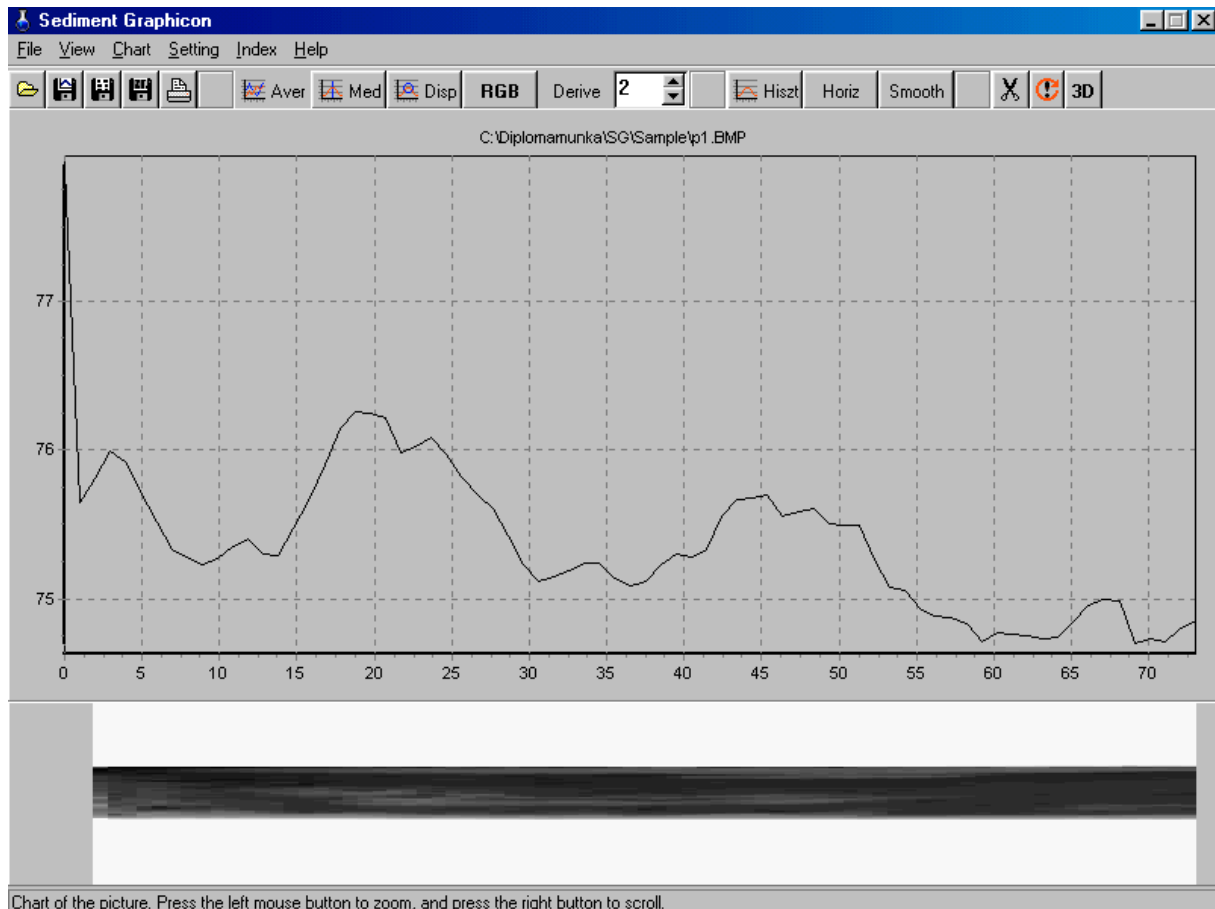
Ritkán, de előfordul, hogy a kép színes és a színek jól elkülöníthetők. Ekkor a RGB gomb hatására a három fő színösszetevő (vörös, zöld, kék) görbe mind külön megjelenik (14. ábra). Érdeemes egymáshoz hasonlítani, és megfigyelni, hogy mikor váltanak, és hogyan aránylanak egymáshoz.



14. Ábra: A színes kép RGB görbéinek a megjelenítése. A színátmenetekenél a görbék helyet cserélnek.

Van olyan eset, amikor a lefotózott minta szélét kitakarjuk. Ez lehet világos, ha sötét a mintánk, vagy éppen fordítva. A lényeg az, hogy jól elkülönüljön a vizsgált minta színárnyalatától.

A fentihez hasonlóan olvassunk be egy p1.bmp nevű fájlt, amely a minta könyvtárban található, és ilyen kitakarással készült. Jól látható, hogy a minta a kép közepén fut végig, két szélén világos kitakarással (15. ábra). A képből készült adatok pedig hibásak, mert a kitakarás pontjait is beleszámolja a valódi minta pontjaihoz.



15. Ábra: A képen egy kitarakt minta képe és átlag görbéje látható.

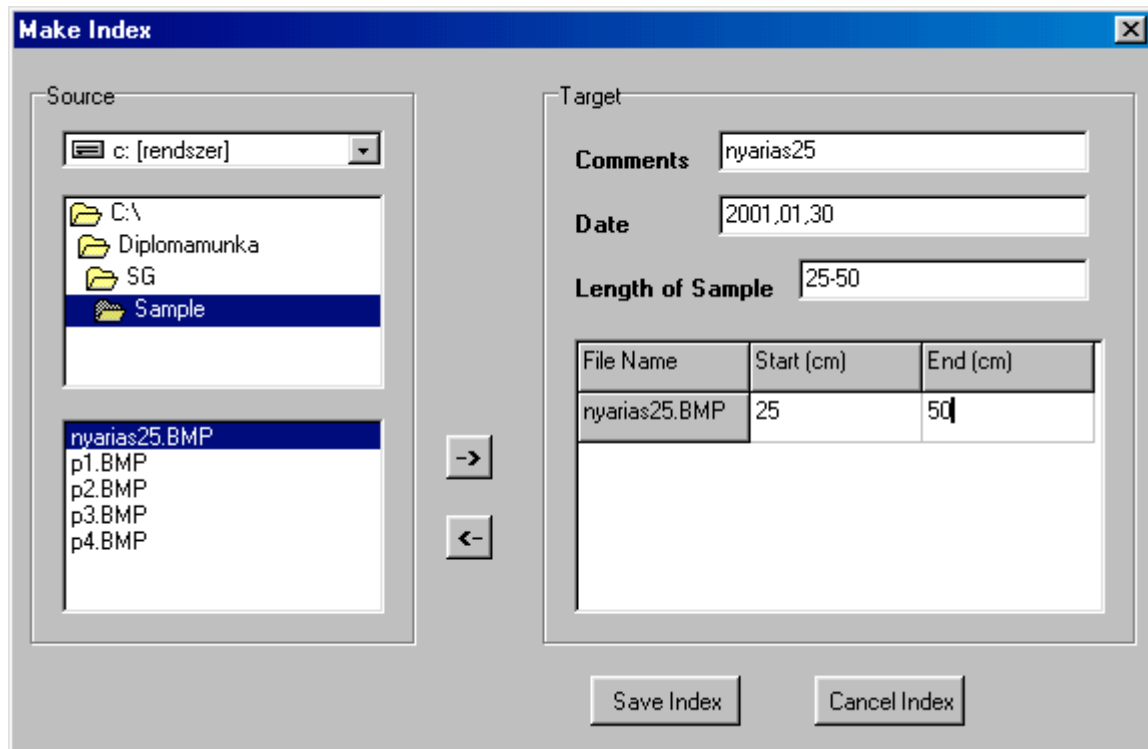
Ezt a problémát orvosolja az AUTO CUTTING PICTURE kapcsoló. Ha ezt bekapcsoljuk, megkeresi a határvonalat a kitarakt és minta között és csak a valódi pontokkal számol. A különbség az y tengelyen olvasható le, ahol a görbe érzékenysége megnő.

Ne feledjük, ez egy kapcsoló: ha bekapcsoltuk, akkor az ezek után beolvasott képeket vágni fogja, még ha nem is kitaraktással készült.

A grafikon x tengelyén egy egység egy kép pixelméretnek számít, mivel nem kalibráltuk a képet. A kalibráláshoz indexet kell készíteni.

9.1 Az Index fájl készítése képekhez.

Először kalibráljuk az előző képet.



16. Ábra: A Make Index segédablak nézeti képe.

Az Index főmenüben Make Index almenüre kattintva az egérrel egy új ablak nyílik meg (16. ábra). Itt kell beállítanunk a megfelelő paramétereket. Két részre van tagolva az ablak. Bal oldalon a Source (forrás), jobb oldalon a Target (cél). A bal oldali részből kell kitölteni a jobb oldali részt.

Először töltsük ki a megjegyzés (Comments = NYARIAS25) szövegbeviteli mezőt; írjuk be a kép nevét vagy a minta kódszámát, a dátumot (Date = 2001.01.30) és a minta hosszát (Length of Sample = 25-50). Tehát a minta bal széle 25 cm-nél van, a jobb széle pedig 50 cm-nél. A mértékegység nem lényeges.

A Source részen három kisebb ablak található. A legfelsőben a meghajtók, a középsőben a könyvtárak, és a legalsóban a fájlok vannak. Keressük meg a nyárias25.bmp fájlt! Kétféleképpen lehet a Target részre másolni: egyrészt a nyilakkal, amik a két részt elválasztják - értelemszerűen az egyik balra, a másik jobbra helyezi a fájlt -, másrészt ha a kiválasztott fájlra duplán kattintunk az egér bal gombjával, akkor automatikusan átmásolódik.

Itt egy táblázatba kerül, ahol a Start és End paramétereket ki kell tölteni. Itt történik a kép valós kalibrációja. A Start ablakba írjunk 25-öt, az End ablakba 50-et.

Ezzel kész is vagyunk, csak el kell mentenünk a Save Index gombbal. A program felajánl egy nevet, de ezt meg is lehet változtatni. Esetünkben fogadjuk el az index nevet.

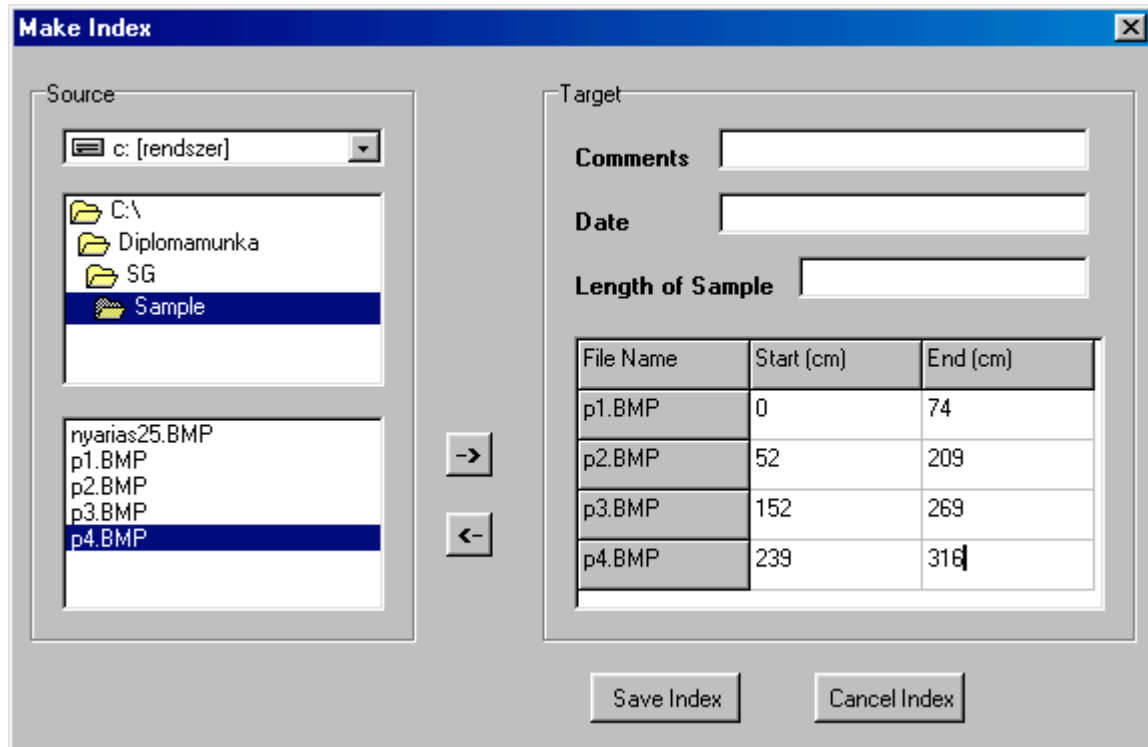
9.2 Index fájlok feldolgozása:

Ugyan a kép kalibrálását elvégeztük a Make Index menüben, de a megjelenítéshez még be is kell ezt olvasnunk. Ezért kattintsunk az Index főmenü Open Index almenüjére. Itt egy Open Index megnyitó ablak jelenik meg. Adjuk meg neki az előbb elmentett INDEX.NDX fájlt, és OK. A program megkeresi a képet és beolvassa azt: a grafikon x tengelye a megadott határok közé esik.

9.3 Több kép összefűzése egy grafikonba.

A programmal lehetőség van olyan képek összefűzésére, amelyek átfedést tartalmaznak, vagy egymás után következnek.

Először itt is index fájl kell készíteni. Make Index ablak megnyitása a menüben (17. ábra). A Target részben a felső 3 ablakot nem fontos kitölteni, csak azonosításra szolgálnak; a mi esetünkben most eltekintünk ettől. Keressük meg a p1.bmp-től a p4.bmp-ig a fájlokat a minta könyvtárban. Másoljuk át ezeket a Target részre, és töltsük ki a táblázatot a következők szerint:



17. Ábra: Több kép összefűzésére szolgáló ablak. A képekhez adjuk meg a kalibrációt.

Miután ezzel megvolnánk, mentjük el a Save Index gombbal az adatainkat, és nevezzük a fájlt p14-nek. Olvassuk be az Open Index menüvel a p14.ndx fájlt. Ekkor sorra fogja venni p1-től p4-ig a képeket. Mindre számol, és egy görbén ábrázolja.

Mivel a minta részei külön lettek lefényképezve, eltolódás léphet fel az átfedő részekben, de a görbe alakja hasonló.

Ha bekapcsoljuk az Auto Cutting Picture (automatikus képvágás) kapcsolót, akkor ezen az eltérésen javítani tudunk. Figyelem, ha bekapcsoltuk az automatikus képvágást, akkor mindig újra olvassuk be az index fájlt, hogy minden képen elvégezze a vágást.

9.4 Az adatok elmentése:

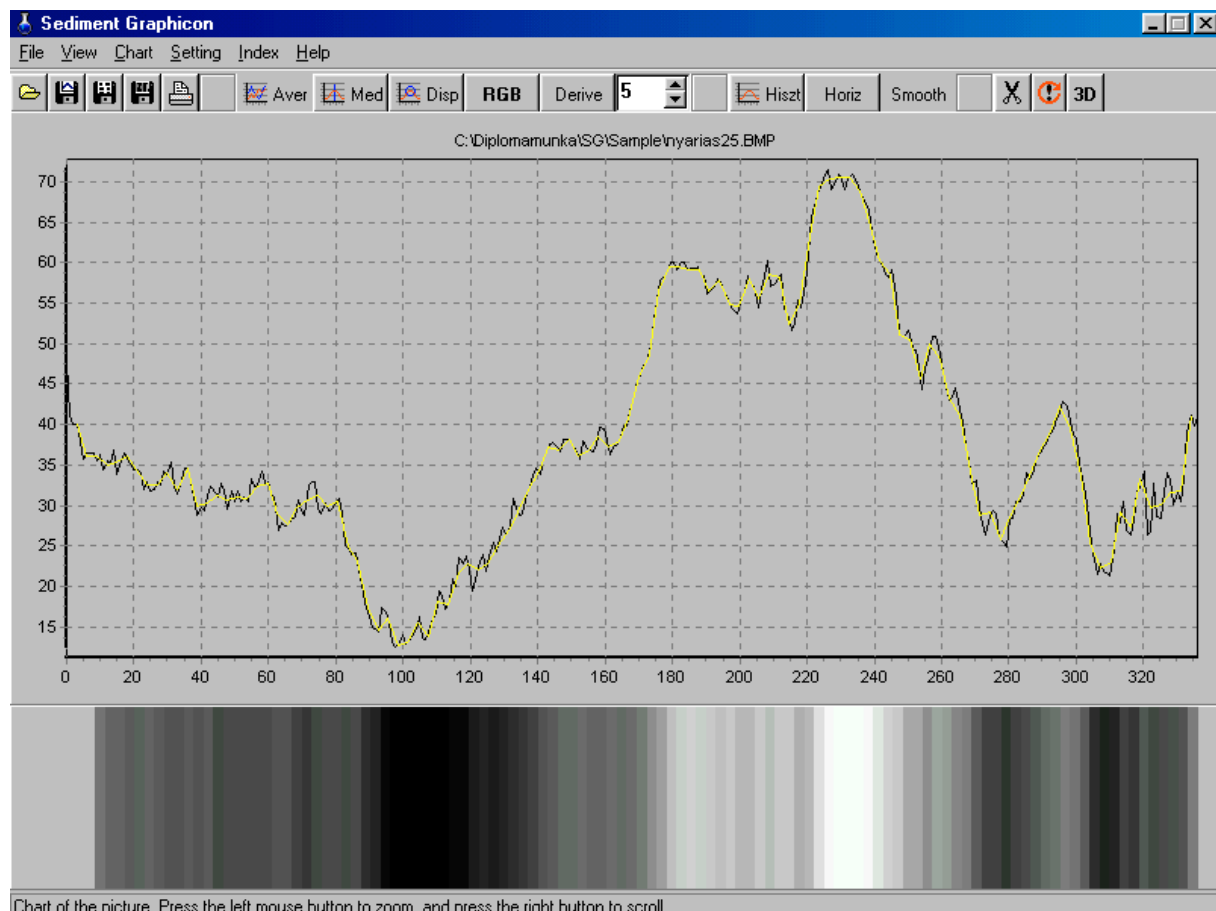
Szükség lehet arra, hogy a program által kiszámolt adatokat nemcsak ábrázoljuk, hanem egyéb számításokat is végezzünk vele. Ezért a programban található egy Save Data menüpont, amivel az éppen látott görbék adatait menthetjük el fájlba. Hívjuk meg ezt a menüpontot, és mentsük el az adatokat. Próbáljuk meg például EXCEL táblázatkezelővel beolvasni (18. ábra). A program dat kiterjesztésű állományba megy, és minden adatpár új sorba kerül. Az adatpárokat szóköz választja el egymástól. Minden elmentett görbének van még egy fejléce is.

	A	B	C	D	E	F	G	H	I
1		This_Is_SediGraph_Data_File							
2		X-Chordinata	Average						
3		0,00	47,79						
4		1,00	41,19						
5		1,99	40,07						
6		2,99	39,99						
7		3,99	37,87						
8		4,99	35,72						
9		5,98	36,52						
10		6,98	36,57						
11		7,98	36,49						
12		8,97	35,56						
13		9,97	36,21						
14		10,97	34,37						
15		11,96	35,53						
16		12,96	35,39						
17		13,96	36,84						

18. Ábra: A Sedigraph program által készített adatok feldolgozása EXCEL táblázatkezelővel.

9.5 Az átlag görbe simítása

Az átlag (Average) görbét egy 5 pontos mozgóátlaggal simítani lehet a Smooth (simít) menüvel. Érdekessége még ezen menüpontnak, hogy a számolt értékek alapján megpróbál egy rekonstruált képet visszaállítani. Ez élesebb mint az eredeti, és spektrumos (vonalas) szerkezetű. Az így készített képet el lehet menteni kép fájlba a File főmenü Save Picture almenüje segítségével (19. ábra).



19. Ábra: A simított görbe illeszkedése az átlag görbére. Lent a számolás útján előállított kép látható.

10 A SediGraph program eltávolítása

A program eltávolítását a számítógépről az operációs rendszer beépített része végzi. Telepítéskor a program regisztrálja magát, és információkat helyez el a gépen. Ezek között szerepelnek az eltávolításhoz szükséges adatok is.

Eltávolítás: Start menü, Beállítások, Vezérlőpult, Programok hozzáadása. Egy ablak jelenik meg, ahol jelöljük ki a SediGraph-ot és kattintsunk a Telepítés/Eltávolítás gombra!

Az ilyen eltávolításnak nagy előnye, hogy a rendszer a regisztrációs adatbázisából törli a programhoz tartozó bejegyzéseket is.

11 A program tesztelése

A program tesztelésére, és a hibák megkeresésére nagy gondot fordítottam. Sok olyan részről nem esett még szó, amely könnyíti a program kezelhetőségét. Ilyen például a könyvtárak kezelése. A program sok megnyitást és mentést tartalmaz, ezért kényelmi szempontból nem elhanyagolható a jó könyvtárkezelés. Ha bármely könyvtárra hivatkozunk, míg meg nem változtatjuk azt, az adott bemeneti és kimeneti könyvtár lesz az aktuális. Ha egy adott könyvtárból beolvasunk egy képet, akkor a mentést is abba a könyvtárba ajánlja fel.

A programot kétféleképpen teszteltem:

1. A számolás sebességét különböző méretű képek esetén különböző gépeken mértem.
2. A beolvasott kép felbontása mennyire befolyásolja a számolt grafikonokat.

Mind a két teszt segítségével a programot optimalizálni lehet.

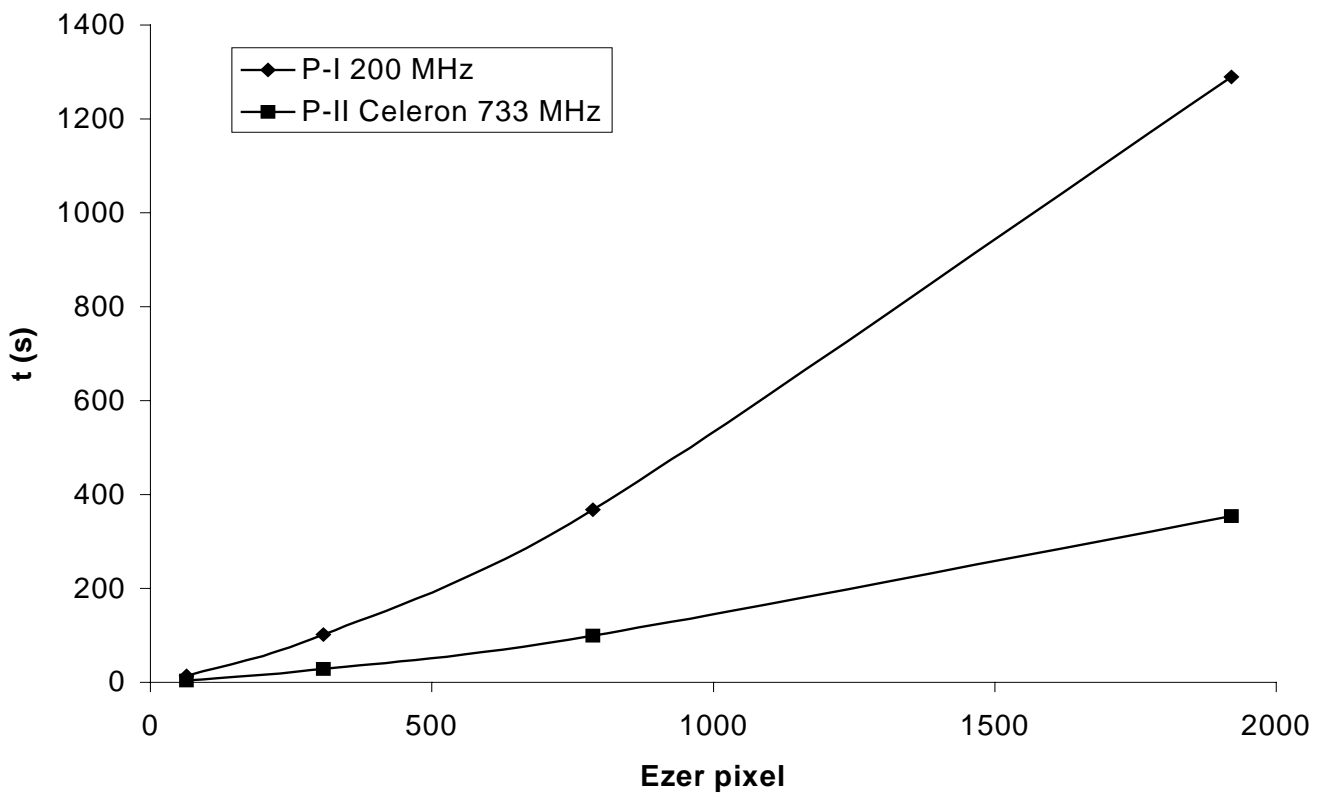
11.1 A sebesség mérése:

Két számítógépen mértem a beolvasott képek számolásához szükséges időt másodpercben. Mind a két esetben WIN98 operációs rendszert alkalmaztam 128 MB SDRAM-al. Az első egy Intel Pentium I-es 200 MHz gép; ez az operációs rendszer szempontjából még éppen elfogadható. A második egy Intel Celeron II-es 733 MHz gép, ami jelenleg jó teljesítményűnek mondható.

A tesztképeket egy képszerkesztő programmal készítettem, mely szürke skálájú volt, és a *2. táblázatban* pixel méretben adom meg a paramétereit. A sebességi adatok a *20. ábrán* grafikusán láthatóak.

2. Táblázat: Mért sebességi adatok a SediGraph programmal.

pixel	P-I 200 MHz, sebesség [s]	P-II Celeron 733 MHz, sebesség [s]
320 x 200 = 64E pixel	14 s	3,6 s
640 x 480 = 307,2E pixel	102 s	28 s
1024 x 768 = 786,432E pixel	368 s	100 s
1600 x 1200 = 1920E pixel	1290 s	354 s



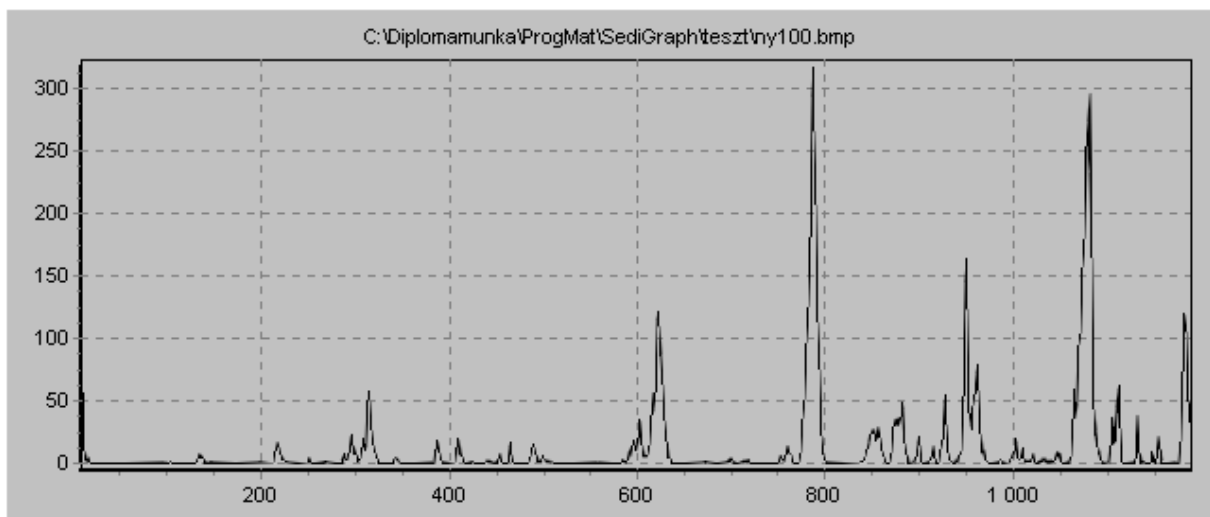
20. Ábra: A sebességek grafikus ábrázolása.

11.2 A felbontás vizsgálata

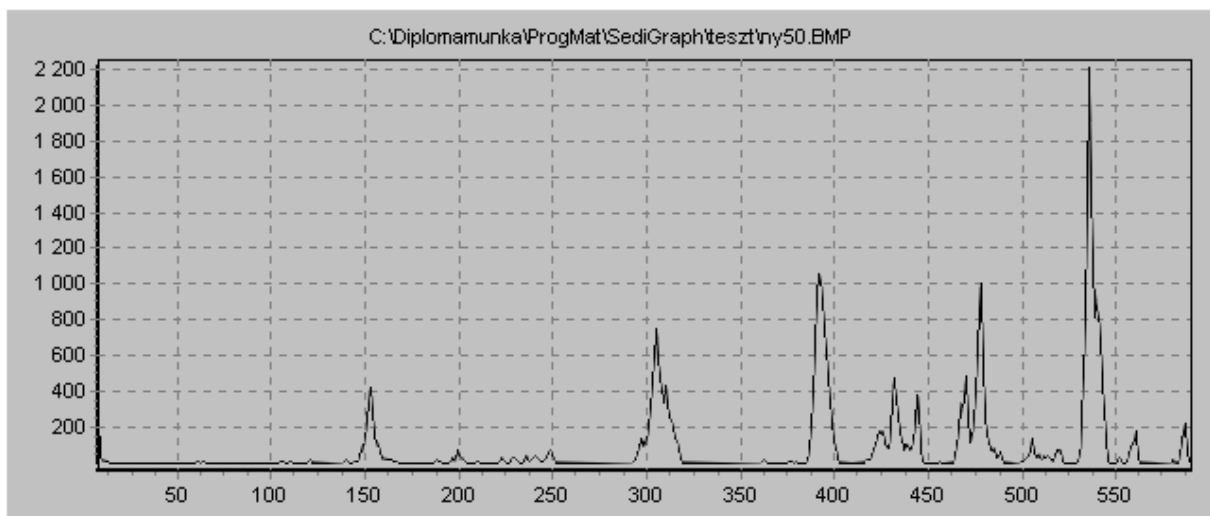
A teszthez egy nagy felbontású képet (1200x400 pixel) vettem 100%-nak. A 3. ábrának megfelelő kicsinyítéseket készítettem belőle egy képfeldolgozó programmal. Ezek után mindet kiértékeltem. A deriválás görbét vizsgáltam, hogy milyen eltéréseket tapasztalok (21-24. ábra).

3. Táblázat: A felbontás vizsgálatánál alkalmazott képméretetek.

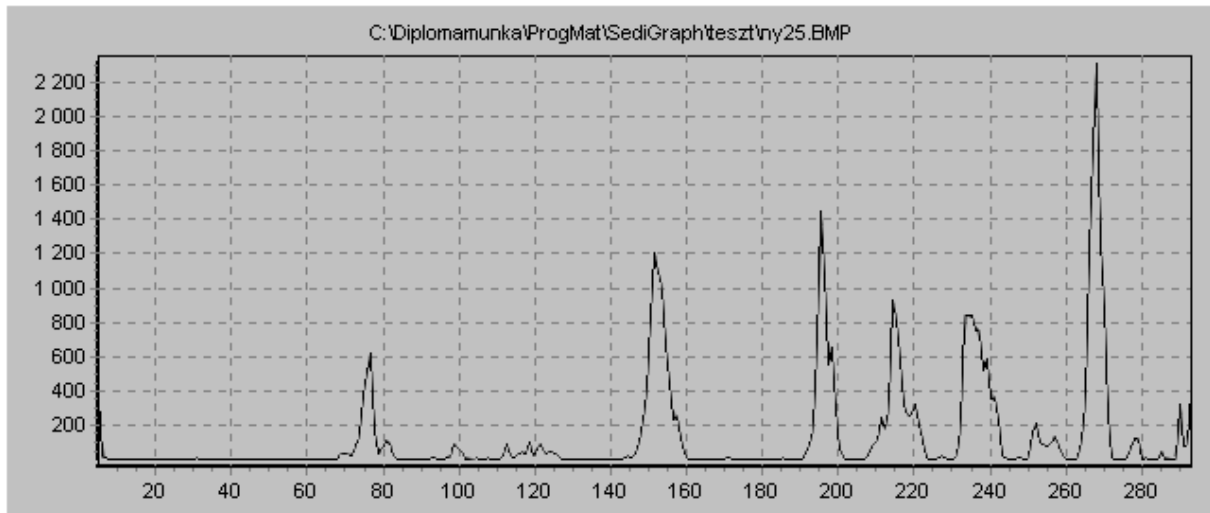
%-os méret	x irányú pixelszám	y irányú pixelszám
100	1200	400
50	600	200
25	300	100
10	120	40



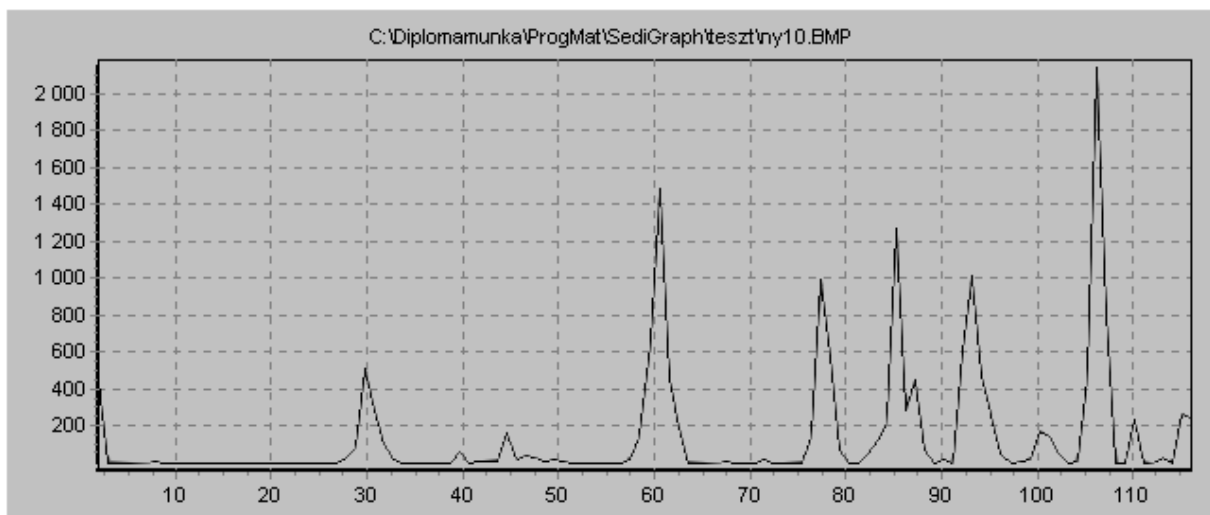
21. Ábra: Felbontás vizsgálata derivált görbével. 100% = 1200 x 400 pixel



22. Ábra: Felbontás vizsgálata derivált görbével. 50% = 600 x 200 pixel



23. Ábra: Felbontás vizsgálata derivált görbével. 25% = 300 x 100 pixel



24. Ábra: Felbontás vizsgálata derivált görbével. 10% = 120 x 40 pixel

A teszt összegzése:

Az első teszteredmény szerint a nagy képek esetén a program exponenciálisan lelassul, ezért célszerű lenne kisebb képekkel dolgozni. Egy 640x480 méretű kép is - ami nem számít „kis” felbontásnak - még elfogadható sebességgel számolható ki.

A második teszt is érdekes eredményt hozott; az eredeti képet tizedrészére kicsinyítve is hasonlóan kiértékelhető grafikonokat kaptunk (21-24. ábra).

A tapasztalat az, hogy a nagy méretű képeket ajánlatos lekicsinyíteni a gyorsabb feldolgozás érdekében, hiszen ezzel nem veszünk jelentős információt.

12 Összegzés

Mivel a program grafikus alkalmazás, ezért a felülete és kezelhetősége is a mai elvárásoknak megfelelően készült. Az ablakok a már eddig megszokott windows rendszerű gombokat, menüket, eszközöket tartalmazzák. A telepítése könnyű, és eltávolítása sem okoz gondot.

A leírás tartalmaz egy oktatórészt is, amely a program használatát példákon keresztül szemlélteti. A program a windows rendszer alapvető ismereteivel könnyen kezelhető, mert új elemet nem tartalmaz, a bemeneti és kimeneti adatok igazodnak az operációs rendszerhez.

A felhasználói segítség (súgó) a már megszokott módon érhető el. Részletes és átfogó segítséget nyújt minden tekintetben.

A programmal réteges szerkezetű talajminták vizsgálhatók, melyekről egy egyszerű kép alapján számos magas szintű információt szerezhetünk különböző matematikai összefüggések használatával. A programmal eddig több száz képet elemeztünk és értékeltünk sikeresen. Képes akár 2000 pont széles képet is ábrázolni a görbéivel, ami közel 20000 pont egyidejű megjelenítése egy grafikonon.

13 Irodalomjegyzék:

- [1] Maurice Tucker: *Techniques in Sedimentology*
- [2] A. R. H. Swan, M. Sandilands: *Introduction to Geological Data Analysis*
- [3] Marcu Cantu: *Delphi 5 (Mesteri szinten)*
- [4] Baga Edit: *Delphi másképp*
- [5] Benkő Tiborné, Kiss Zoltán: *Programozás Borland Pascal 7.0 Rendszerben*
- [6] Dr. Szirmay-Kalos László: *Számítógépes Grafika*
- [7] Windows operációs rendszer: *Súgó*