

# Látványos oktatás egyszerű multimédiás elemek programozásával Delphiben

Menyhárt László Gábor

menyhart@elte.hu  
ELTE IK

**Absztrakt.** A cikkben bemutatok egy ötletes megoldást arra, hogy hogyan lehet egyszerűen és olcsón fizikai kísérletekből adatokat kinyerni, melyekkel már számolást is lehet végezni. Ehhez az interdiszciplináris feladathoz segítségül hívom Delphit, melyben a digitális videót, mint multimédiás elemet tudjuk programozni. Példát mutatok egyszerű multimédiás elemek programozásától az első hatékonysági lépésig a videó feldolgozásában. A bemutatandó forráskód a programozást Delphiben tanuló középiskolai diákok számára is érthető bonyolultságú.

## 1. Bevezetés

Ebben a cikkben szeretném bemutatni, hogy multimédiás elemeket hogyan lehet programozni. Bemutatom, hogy kis ötlettel milyen egyszerűen lehet látványosabbá, élvezhetőbbé és érthetőbbé tenni a természettudományos oktatást.

Már több konferencián tartottam rövid előadást arról, hogy hogyan próbáltam matematika-fizika és informatika szakos tanárként a tárgyaimat illetve a multimédiát összekapcsolni.

Kutatásaim során több olyan alkalmazást, segédprogramot, internetes oldalt találtam, amik segítik az oktatást. Ezek közül most néhány olyat emelnék ki, melyek az alapötletet adták. A WPtools [3] nevű segédprogram a Microsoft Excel-be beépülve segíti az adatfeldolgozást, az adatábrázolást és függvényillesztést. Használata bármilyen természettudományos tantárgy gyakorlásánál lehetséges, ahol több adat szerepel illetve adatsorokban jelenik meg a szükséges információ. A Logger Pro [2] alkalmazás az adatokat saját táblázatban kezeli, tud függvényt illeszteni (curve fit). Az adatok felvételét nagyon jól támogatja, több eszközzel tud kapcsolatot teremteni, például LabPro, CBL 2 illetve Texas Instruments graphing calculators. Ennek az ára \$149. Nekem mégis a VideoPoint nevű szoftver keltette fel legjobban az érdeklődésemet, ugyanis ehhez nem kell kézzel bevinni az adatokat egy táblázatba, nincs szükség külön hardware termékekre az adatfelvételhez, mert ez digitalizált videofelvételek analizálásával jut adatokhoz, és a saját táblázatos adatkezelőjében tud statisztikákat készíteni, függvényt illeszteni. Ennek azonban megvan az ára. A tanári csomag (VideoPoint Teacher's Kit with Capture) \$225, hogy 5 diák is használhassa akkor \$585.

Ebben a cikkben egy, ez utóbbihoz hasonló, saját megoldást készítek el. Az alapoktól kezdve bemutatom, milyen multimédiás elemeket lehet a Delphi-ből kezelni.

## 2. Kérdés felvetések

Hogyan lehet érdekesebbé tenni az oktatást? Kevés az idő? Nincs lehetőség kísérletezni fizikaórán? De. Azt is kell. Mégis mi a megoldás, ha mindenkinek más és más eredmény jön ki? A matematikaórán megoldott szöveges feladatok csak kitalációk, vagy mennyire kötődnek a hétköznapi problémákhoz? Informatika órán csak szövegszerkesztést és táblázatkezelést tanulják a diákok? A programozás, ahol tanítják, csak billentyűzetről bevitt adatokkal, legjobb esetben is csak fájlokból szerzett adatokkal dolgozik?

## 3. Megvalósítás

Ezekre és még sok hasonló kérdésre ad egy megoldást a digitális technika, hiszen ennek a fejlődésével a tanórákra nem csak a televízióban levetített filmek használhatóak, nem csak a saját videofelvételünket nézhetjük meg, hanem ezeket ki is elemezhetjük számítógép segítségével. Ebben a cikkben nem foglalkozok az adatok elemzésével, statisztikák készítésével, függvényillesztéssel. Megelégszem azzal, hogy sikerül az adatsorokat numerikus formában megjelenítenem a számítógépen.

### 3.1. Kreatív oktatás

#### 3.1.1. Matematika

A matematikaoktatásban egyre több a szöveges feladat az értő olvasás, mint alapkompétencia megszerzése érdekében. A szöveges feladatok támogatása "valódi" adatokkal talán még jobban segítheti a megértést. Talán a diák motiváltsága is érdeklődéssé, kíváncsisággá vagy szórakozásává változhat.

#### 3.1.2. Fizika

A fizikaoktatás szerintem kihagyhatatlan része a kísérletezés. Ha a képletek, összefüggések megtanulása mellett számolásra van idő, miért ne mérhetne mindenki? Legalább virtuálisan. Így jól láthatóvá és érthetővé válik a hiba, mint fogalom. Mindenki saját adatokkal dolgozhat, számolhat, és várhatóan ugyanarra az eredményre jut bizonyos hibán belül.

#### 3.1.3. Informatika

Az algoritmikusgondolkodás, a programozás, a projekt szemlélet, a csapatmunka elsajátítása konkrét, valódi problémák megoldásával olyan feladatokká válhatnak, melyet nem a kényszerűségből kell elvégezni, hanem igazi célja van és ez által kihívássá, szórakozássá válhat a tanulás.

#### 3.1.4. Kreatív együttműködés

A multimédia segítheti a kreatív együttműködést. A mindennapi eseményeket videofelvételről megtekinthetjük és kiértékelhetjük. Ha hétköznapi jelenségeket figyelünk meg videofelvételről, akkor ezzel személyessé, testközelivé válhatnak a jelenségek és magyarázataik, és talán jobban felkeltik az érdeklődést, mint az olvasott szöveg. Az informatika, fizika és matematika együtt-

működésétől azt várnám, hogy tegye érdekesebbé az órákat, érthetővé a tanultakat és kapcsolja össze a hétköznapi valóságot az iskolában tanultakkal.

### 3.2. Egy ötlet a határterületről

A három tantárgyam határterületéről a legkézenfekvőbb ötlet az, hogy dolgozzunk fel egy fizikai kísérletet. A végeredmény legyen egy függvény, melyet egy pontsorozatunk. Az adatok megszerzésében pedig hívjuk segítségül az informatikát, illetve azon belül a multimédia programozását. Most arra koncentrálok, hogy a jelenségtől az adatok megszerzéséig hogy juthatunk el gyorsan és látványosan.

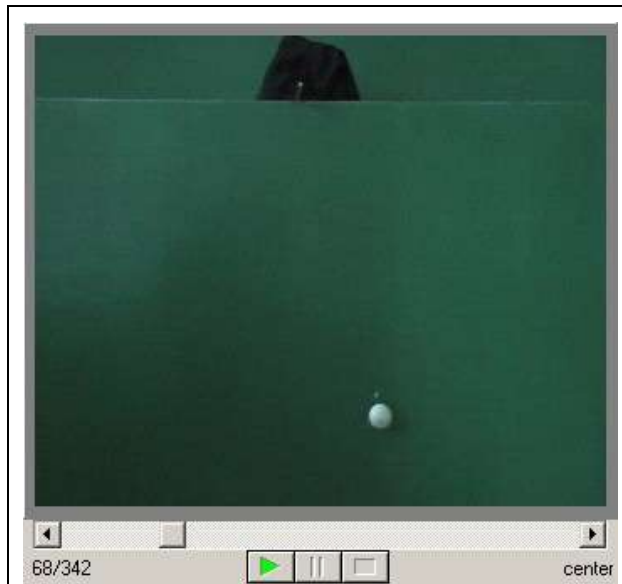
#### 3.2.1. Videó digitalizálás

Egy videokamera segítségével a megfigyelt hétköznapi eseményt rögzíteni tudjuk. Ha ez egy modernebb digitális kamera, akkor nagyon egyszerű ezt a felvett anyagot a számítógépre menteni, hiszen már nem kell a digitalizálással bajlódni, csak az anyag lementését kell elvégezni. Ezt támogatják a kamerák és a számítógépek. Ingyenes szoftvereket lehet beszerezni a könnyebb munka érdekében. Ilyenek például a következők: VideoInspector, Virtual Dub.

Ha régebbi típusú analóg videokameránk van, akkor egy TV-kártyára is szükség lesz, hogy a számítógép által ismert formátumba be tudjuk digitalizálni az anyagot. Egy olyan számítógép, amelynek a felszereltsége és teljesítménye megfelelő az ilyen digitalizálásra elérhető minden iskolában.



1. ábra: Régebbi felvétel



2. ábra: Újabb felvétel

Videóra vettem két ingaórát, az elsőt egy hagyományos videokamerával, majd bedigitalizáltam; a másodikat digitális videokamerával. A videók elkészítésénél, és ha szükséges, digitalizálásuknál nagyobb feladat inkább az, hogy a számítógépen lévő digitalizált formátumú videó anyagból, hogyan tudunk számsorozatokhoz jutni.

### 3.2.2. Adatfelvétel

A videó nem más, mint képkockák sorozata. Egy másodperc alatt 25 képkockát látunk, vagyis 1 képkockát 40 ms-ig. Tehát az idősorozatunk adott, olyan számtani sorozat, aminek a diszkriminánása 40 ms.

Már csak az  $X$  és  $Y$  koordinátákhoz kell hozzájutnunk. Szerencsére a videót képkockánként tudjuk léptetni megállítva az időt, így egyszerűbb képfeldolgozási problémára tudjuk visszavezetni az  $(X, Y)$  koordináták kinyerését.

A kézi adatfelvétel az első, ami eszünkbe juthat. Készítsünk egy kis programot, ami a videót képkockánként tudja léptetni a videót. A léptetést egérekattintásra végezze el a program, hogy az egér pozícióját,  $(X, Y)$  koordinátáját elmenti. Így a videón látható célobjektumra kattintva, annak 40 ms-ként változó pozícióját el tudjuk menteni. Tehát a videó anyag végére érve kapunk három számsorozatot, a 40 ms lépésközű időket és a hozzá tartozó  $X$  illetve  $Y$  koordinátákat.

Ennek elkészítése nagyon egyszerű. A kattintgatások interaktív közreműködést igényelnek. A diákok magukénak érezhetik az adatokat. Előny még, hogy egyszer-kétszer direkt mellékattintva is kijöhet az eredmény. Jól bemutatatható, ábrázolhatóak a hibák, megérthető a hibakezelés, hibaszámítás lényege.

### 3.3. Megvalósítás

Lássuk, hogy egy ilyen program megírásához milyen eszközeink vannak, hogyan kell elkezdenünk. Először végigveszem, hogy milyen multimédiás elemek vannak és ezeket, hogyan támogatja a Delphi. Ezután részletesen foglalkozom a videók kezelésével.

#### 3.3.1. Multimédiás elemek Delphiben

A multimédia alkotóelemei lehetnek szövegek, állóképek, hangok, animációk, videofilmek. Ezek az alkotóelemek együttes használatával még több információt közölhetünk. Egy néma videofilm és egy adott időpontban megszólaló sípszó nem csak két különálló multimédiás elem a saját információjával, hanem többet közölhet. Ahol szól a síp, az fontos része a videónak. Ezért mondjuk azt, hogy a multimédia kvalitatív tartalmú fogalom. „1 videó információ”+ „1 audió információ” > „2 információ”.

A Delphi egy programozási környezet, melyet a Pascal nyelvre építve, azt kibővítve hoztak létre. Segíti a Windows-os standard alkalmazások készítését. Támogatja a multimédiás elemeket.



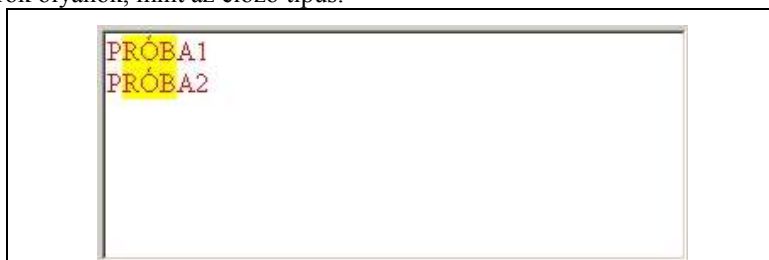
3. ábra: Szöveges mező - TEdit

A szöveges adatok megjelenítését több objektum is segíti. Az egyik legegyszerűbb lehetőség csak egy szöveges adatsor beolvasására a TEdit típus. Ennek segítségével rövidebb, egyszerű szöveget tudunk beolvasni. Az Entert nehéz kezelni vele.



4. ábra: Több soros szöveges mező - TMemo

Kicsit összetettebb típus a TMemo, ami nagyobb, több sorból álló szöveg kezelésére is jó. Itt az egyes sorok olyanok, mint az előző típus.



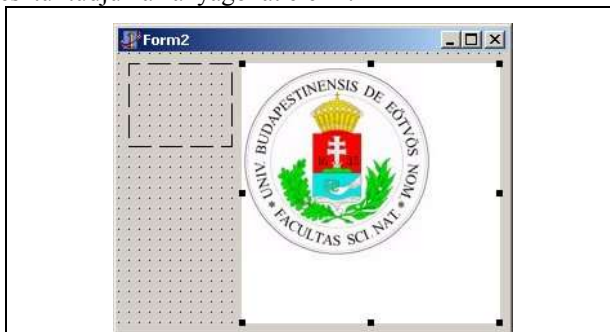
5. ábra: RTF forrás kezelése - TRichEdit

A következő típussal már nem csak szöveg, hanem annak különböző tulajdonságai is megadhatóak. A betűk méretét, színét és egyéb tulajdonságát, sőt bekezdéseket is meg lehet adni a TRichEdit használatakor. Ez a komponens Microsoft RTF formátumát használja.



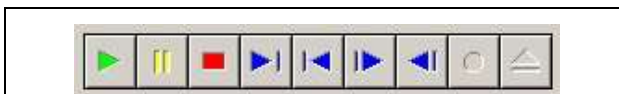
6. ábra: HTML megjelenése - TWebBrowser

A legszélesebb körű megoldás a képek, hangok, szövegek, animációk együttes alkalmazására az Internet nyújtotta lehetőség a HTML segítségével. Szerencsére a Delphi tud a TWebBrowser segítségével az alkalmazásba beágyazni egy Microsoft Internet Explorer-t, így teljes funkcionálisú böngészőn keresztül tudjuk az anyagokat elérni.



7. ábra: Képmegjelenítő komponens - TImage

A képek megjelenítését, kezelését a TImage típus segíti. Ezzel lehet állóképet megjeleníteni egy alkalmazás felületén. A TImage objektumok rá kell helyezni a form-ra, mint ahogy minden alapelemet erre kell helyezni. Be kell állítani a méretét, amit egy fekete szaggatott keret jelez, majd be kell olvasni a megjelenítendő képet. Ezt el lehet végezni a fejlesztési időben és a futási időben is.



8. ábra: Médiaelemek kezelése - TMediaPlayer

A TMediaPlayer típus segítségével pedig audió és videó anyagok lejátszására van lehetőség. Az ábrán látható kezelőfelület jelenik meg a programunkban, amivel futás közben tudjuk az anyagot elindítani, megállítani, léptetni. A videó megjelenésének helyét a típus Display nevű tulajdonságának (property) kell megadni.

```
MediaPlayer1.Display:=Panel1;
```

### 3.3.2. Egy videó képkockáinak kezelése

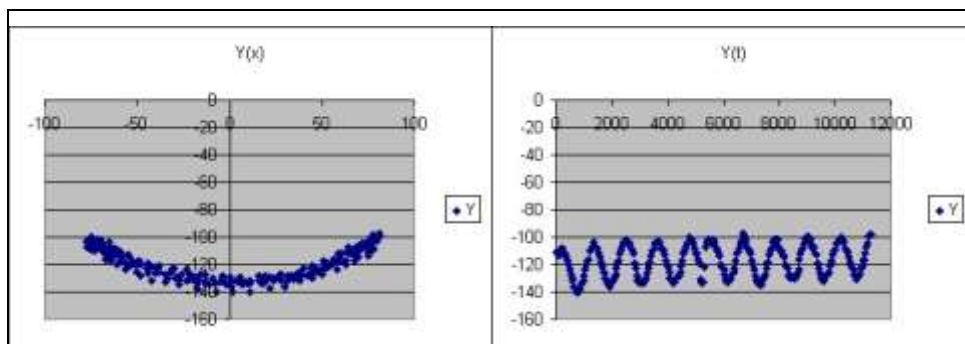
A TMediaPlayer típus Position property-je a videók esetében megadja, hogy hányadik sorszámú képkocka az aktuálisan kirajzolt, még akkor is, ha nem fut a videó lejátszása. A videó képkockáinak kirajzolási helye pedig lehet egy TPanel típusú felület az alkalmazáson.

Ezen a Panelen való kattintást a TPanel OnClick eseményével le tudjuk kezelni, és előtte még az egérpozíciót is el tudjuk menteni az OnMouseDown esemény segítségével. Ezzel az ismerettel már minden adott, hogy az egérekattintásokkal történő adatfelvételhez szükséges programot elkészítsük.

X (156)	Y (39)
204	224
206	225
206	227
203	228
198	229
193	230
189	230
186	231
176	231
167	233
162	235
153	235

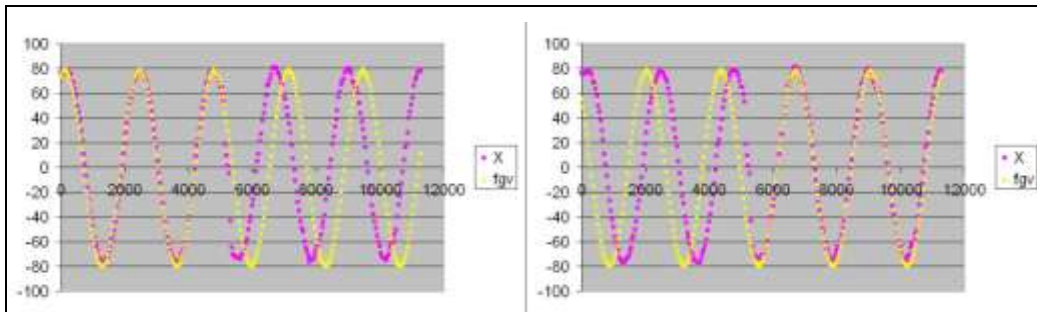
9. ábra: Adatok

Így az adatokat, az egérpozíciók X illetve Y koordinátáit el tudjuk menteni egy táblázatba. Az ábrán a fejlécben, zárójelben látható értékek az inga középpontjának koordinátáit tartalmazzák.



9. ábra: Adatok

Ezeket az értékeket el tudjuk menteni egy tabulátorral elválasztott adatokat tartalmazó szöveges fájlba, amit Excel-be be tudunk olvasni. Majd ott fel tudjuk dolgozni őket.



9. ábra: Adatok

A régebbi videó digitalizálása során egy kis ugrás került a videóba, mert a videó folyamatosan futott, de amíg a lassabb számítógép a puffertelt adatok mentését végezte, addig elveszett pár képkocka.

$$X(t) = A \cdot \sin\left(\frac{2 \cdot \pi}{T} \cdot t + \alpha\right) \quad (1)$$

A 9. ábrán a lila koordináták a videón látható adatok, a sárga koordináták pedig számolt értékek az (1) képlet szerint. A két eltolás ( $\alpha$ ) különbségéből kiszámolható, hogy 10 képkocka maradt ki a mentés alatt. A  $T$ -re pedig 2,33575662 másodperc jött ki, vagyis a hiba 17%, ami elég nagy, de ha figyelembe vesszük a hibaforrásokat, akkor ez elfogadható véleményem szerint. Hibaforrás ugyanis elég sok helyen lehetett: a régi videó nem állványról, hanem kézből készült; annak digitalizálásánál már láttunk hibát; az adatokat, nem feltétlenül a súlypont eltalálásával, eger kattintgatásokkal vettük fel; az excelben lévő képlethez az értékeket „szemre” állítottuk be, hogy jó legyen a grafikonokon a takarás.

### 3.3.3. Hatékonyabb adatfelvétel - nyomkövetés

Persze az emberi beavatkozás lassítja az adatok felvételét. Hiszen ha csak 10 másodperces videó anyagból szeretnénk kinyerni az adatokat már 250-et kell kattintanunk. Ez elsőre tényleg jó lehet, mint ahogy az előző fejezetben írtam, de hosszú távon lassíthatja a munkát. Ezért célszerű lenne, egy olyan lehetőséget is beépítenünk a programunkba, hogy az adatok felvételét a gép is segíthesse. Az objektum helyének meghatározásához két lehetőség is lenne. Az egyik megközelítés szerint az objektum éleit határoznánk meg, míg a másik szerint a teljes objektumot próbálom egyben kezelni. A súlypontszámítást azért választottam, mert így a raszteres képen nem kell élkimelés, majd az éllel megadott esetleg konkáv objektum középpontját számolni, hanem a bármilyen alakú objektum egyes pontjai segítségével keressük meg a súlypontot.

Az ötletem az, hogy az egyes képkockákból készítsünk egy fekete-fehér ábrát, ahol a kép nagy része fehér a célobjektum pedig fekete. Majd a képnek számoljuk ki a „súlypontját”, vagyis határozzuk meg, hogy a fekete pontoknak hol van a középpontja. Itt tehát azzal a feltevéssel éltem, hogy a megfigyelt kétdimenziós test jó közelítéssel homogén és középpontosan szimmetrikus. Még finomítható ez a nyomkövetés, ha nem az egész képet dolgozzuk fel, hanem csak a célobjektum kis környezetét. Mégpedig azt a környezetet, ahova az objektum 40 ms alatt eljuthatott, ami nagy valószínűséggel nem a teljes kép.



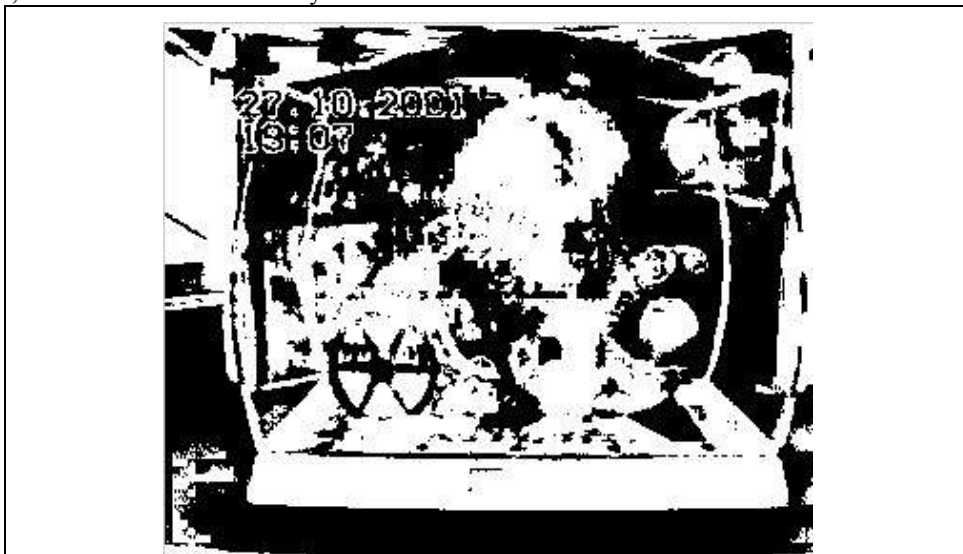
A következő kis trükk segítségével pedig az egyes képkockákat, mint képeket is el tudjuk érni.

```
Type
TPanel = class(ExtCtrls.TPanel)
public
    property Canvas;
end;
TForm1 = class(TForm)
    MediaPlayer1: TMediaPlayer;
    ...
    Panel1: TPanel;
    ...
end;
```

Ha használjuk a fenti kódot, akkor a programunkban a panel rajzvászonját, Canvas-át is elérhetjük, vagyis pixelenként fel tudjuk dolgozni a panelre éppen kirajzolt képeket. A kirajzolt kép (I,J) pozícióján lévő képkocka színét a következő paranccsal tudjuk megkapni.

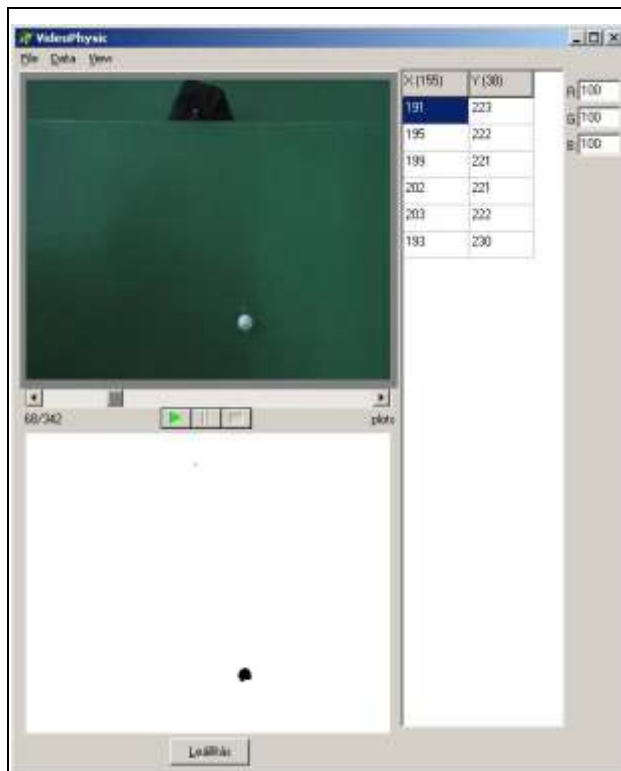
```
Panel1.Canvas.Pixels[i, j];
```

Először egy fekete-fehér szűrést kell a képen elvégezni, majd a kép súlypontját ki kell számolni. A képet pixelenként feldolgozzuk és a kép minden képpontjához egy 0-t vagy 1-t rendelünk. A megfigyelt objektumhoz 1-t a többihez pedig 0-t. Ezt legkönnyebben úgy tudjuk megtenni, hogy sötét háttér előtt világos a megfigyelt objektum, vagy fordítva. Így pixelenként meg tudjuk mondani az RGB színösszetevőkből, hogy az, az objektum vagy sem. A példafelvételinken akkor volt az objektum és lett a pont értéke 1, rajzoltunk ki fekete pontot, ha az RGB színösszetevők mindegyike nagyobb volt egy paraméterként megadható értéknél, vagyis fehérebb volt, mint a beállított szürke árnyalat.



10. ábra: Fekete-fehér szűrés

Az első felvételnél még nem használható ez a szűrés, mert nem figyeltünk arra, hogy az objektum és a háttér elég kontrasztos legyen, de a laboratóriumi felvételünk már megfelelt ennek a követelménynek is.



7. ábra: Nyomkövetés szűréssel a kész programban

#### 4. Eredmények

Tudom, hogy még nagyon sok továbbfejlesztési lehetőség van, nem egy teljesen készterméket mutattam be, hanem egy ötletet, lehetőséget, mely kis kreativitással és ügyességgel nagyon jól hasznosítható lehet az oktatásban. Olcsó, érdekes és könnyen megvalósítható a fejlődő és kurrens technológiák felhasználásával. A bemutatott interdiszciplináris tananyagok, a forráskódok és programozás-technológia szempontjából a cikk egy középiskolai szakkörön feldolgozható és továbbgondolható. A videókat, a forráskódot és a kiértékeléshez használt Excel táblázatot a következő linken elérhetővé tettem.

<http://xml.inf.elte.hu/konferencia/2008/INFODIDACT>

#### Irodalom

1. VideoPoint <http://www.lsw.com/videopoint/> és [http://www.pasco.com/software\\_prod/home.html](http://www.pasco.com/software_prod/home.html)
2. Logger Pro 3 <http://www.vernier.com/downloads/lp3demo.html>
3. WPtools [http://physics.dickinson.edu/~wp\\_web/wp\\_resources/Documentation.html](http://physics.dickinson.edu/~wp_web/wp_resources/Documentation.html)