



SPECTACULAR EDUCATION WITH PROGRAMMING PLAIN MULTIMEDIA ELEMENTS IN DELPHI

László Menyhárt

Abstract: I present a witty solution in this article about how we may obtain data simply and cheaply from physical experiments, with which we can do counting, as well. To this interdisciplinary task I turn to Delphi for help, in which we can program a digital video as a multimedia element. I show an example from the programming of plain multimedia elements until we get to the first efficient step in the processing of the video. High-school students who are learning programming in Delphi are able to understand the presented source code because it is not too complicated.

Keywords: programming plain multimedia, Delphi, physical experiments.

1. Introduction

I would like to demonstrate in this article how we can program multimedia elements. I demonstrate how simply we could make the scientific education more spectacular, more enjoyable and more understandable with a small idea.

I have already made short presentation on more conferences about how I tried to connect my subjects of mathematics, physics and informatics with the multimedia.

In the course of my research I found more applications, utilities and web pages which help in education. I would emphasize some of those now which provided the basic idea. WPtools [3] utility helps in the data processing, the data depiction and function joint integration into the Microsoft Excel. We can use this utility to practice any kinds of scientific subjects where there are more data or the necessary information appears in data lines. Logger Pro [2] application handles the data in its own table and it can make curve fitting. It provides good features to the uptake of the data, it can create a contact with devices, for example LabPro, CBL 2 or Texas Instruments graphing calculators. The price of this is \$149. Despite all these, I became most interested in a software named VideoPoint because using it, it is not necessary to take the data into a table by hand, there is no need of separate hardware products to the data registration, because this utility gets data by analysing digitized video recordings, and it can prepare statistics in its own tabular data handling system, and it can make curve fitting. But then again it is more expensive. The teacher's pack (VideoPoint Teacher's Kit with Capture) costs \$225, and the version which allows 5 students to use it is \$585.

In this article I will introduce my own solution that is similar to the latter. I present from the basics what kind of multimedia elements may be treated with Delphi.

2. Questions arising

How is it possible to make education more interesting? Do we have too little time? Are there no opportunities to make experiments on a physics lesson? Yes, there are! Of course, experiments are needed. But what is the solution, if everybody gets different results? Are the text tasks solved on mathematics lesson only fictions, or are they connected to everyday problems? If they are, to what extent? Do students study only word processing and spreadsheet programming on an informatics lesson? Does the teaching of programming work with data simply inserted with a keyboard or others coming from files?

3. Realisation

Digital technology gives a solution to these and many similar questions. With its development not only can the films on television be used or can we watch our own video recordings on the lessons but we are also able to analyze these with the help of a computer. This article is not about analysing data,

making statistics and curve fitting. Now I content myself in displaying the data in a numeric form on the computer.

3.1. Creative education

3.1.1. Mathematics

There are more and more text tasks in the mathematics education to practice understanding reading as a basic competence. Understanding might be helped much better by supporting the text tasks with real data. Perhaps students' motivation may change to interest, to curiosity or fun.

3.1.2. Physics

In my opinion the experimentation cannot be omitted as a part of physics education. Why could not everybody make measurements if they must learn formulas and contexts and there is sufficient time for counting on the lessons? Students might at least do it virtually. In this manner the concept of error can become visible and understandable. Everybody can work and reckon with their own data and probably they can get the same result within margin of error.

3.1.3. Informatics

By solving real problems, acquiring algorithmic thinking, programming, project view and teamwork can stop being a necessary evil and start to have real goals. In this way experiential learning can become a challenge and fun.

3.1.4. Creative cooperation

Multimedia can help creative cooperation. Daily events can be watched from video recordings and we may evaluate them. If we observe everyday phenomena from video recordings, these and their explanations may turn personal and they might raise interest more effectively than written texts. I expect from the cooperation of informatics, physics and mathematics that it makes lessons more interesting, the material more understandable and that students' studies can connect more to everyday reality.

3.2. An idea from the borderline

An idea from the borderline of these three subjects is evaluation of a physical experiment with the help of video recording. Let the final result be a series of data points. We turn to informatics or, more specifically, the programming of multimedia for help in getting the data. Now I concentrate on how we can get data quickly and spectacularly.

3.2.1. Video digitization

We can record the observed everyday event with a video camera. If this video camera is a modern digital one, the recorded film can be saved easily on a computer. We should copy the file to the computer and we do not have to digitalize it yet. This is supported by cameras and computers. There are free and open-source softwares for easier work, for example VideoInspector, Virtual Dub.

If we have an older analogous video camera, we need a TV card to digitize the film into a computer-known format. A computer able to do this, with enough resources and power is available in every school.



Figure 1: The 67th frame of an older uptake

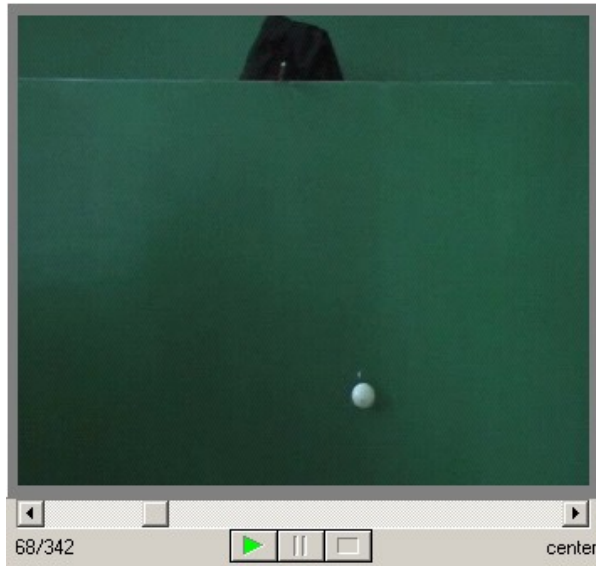


Figure 2: The 68th frame of a newer uptake

I recorded two pendulum clocks. The first one (Figure 1) happened with a traditional camcorder. The second one (Figure 2) happened with a digital camcorder. The completion of the videos and their digitalization are not so difficult tasks as the more complicated issue, retrieving the data from the film.

3.2.2. Data registration

A video consists of series of frames. We can see 25 frames in a second, so one frame is visible by 40 milliseconds. Thus, the time series is given, it is an arithmetical sequence with the discriminant 40 ms.

With this we have to retrieve just X and Y coordinates. Fortunately, the video can be stepped by frames with stopping the time so we were able to reduce the extraction of (X,Y) coordinates to an image processing problem.

The first idea is the manual data registration. Let us create an application that can make the video step by frames. This tracking can be controlled by mouse clicking, and, at the same time, the application can save the position of the cursor with the (X,Y) coordinates. By clicking on the target object we can save the different positions every 40 ms during the whole video. So we can obtain three series of numbers by the end of the video. We will have the 40 ms times, and also the X and Y coordinates.

The completion of this is very simple. Clicking requires interactive collaboration. Students may feel that the data are their own. Another advantage is that the result may also be calculated with one or two clicks to wrong places. The errors can be presented easily and the point of handling and calculating errors can be understandable.

3.3. Implementation

Let us see what kind of tools we have to create an application of this kind and how we can start it. At first I check the multimedia elements and their supporting by Delphi. Then I deal with the handling of videos in detail.

3.3.1. Multimedia elements in Delphi

The components of the multimedia may be texts, pictures, sounds, animations and videos. We can give more information with the usage of these components together. A dumb video and a whistle can mean more than two separate pieces of simple information. The most important part of this video is where the whistle can be heard. That is why we can say that multimedia is a notion with qualitative content. "1 pieces of video information" + "1 pieces of audio information" is more than "2 pieces of information".

Delphi is a programming environment which is based on the Pascal language and which was extended. It helps creating standard Windows applications. It supports the multimedia elements, too.



Figure 3: TEdit is the text input field

More objects manage the representation of the text. One of the most popular opportunities is the TEdit type (Figure 3) which is good for reading text data. It is useful for shorter, plain texts. Handling of the Enter is not easy with this.

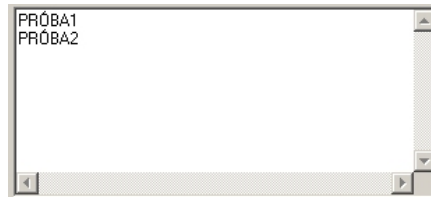


Figure 4: We may handle more rows with TMemo

TMemo (Figure 4) is a bit more complex than the previous one. It is good for handling larger texts and more rows. Each row is similar to the previous type.

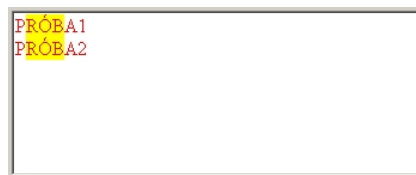


Figure 5: TRichEdit helps with RTF treatment

The next type can contain not only the text but also its different characteristics. By using TRichEdit (Figure 5) we can define the size of letters, their colour, their other characteristics and it is possible to use paragraphs. This component uses Microsoft RTF format.



Figure 6: TWebBrowser displays the HTML contents

The solution of HTML supports the simultaneous management of texts, pictures, sounds and animations. Fortunately Delphi, with the help of TWebBrowser (Figure 6), can embed a Microsoft Internet Explorer into the application, which helps me to reach data through a browser with full functionality.

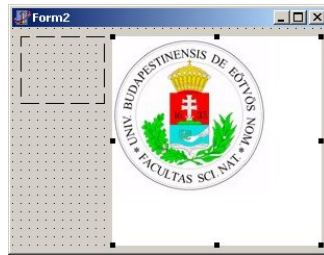


Figure 7: TImage component displays images

The representation of the pictures is helped by the TImage type (Figure 7). It can display a still picture on the surface of an application. TImage objects need to be placed onto a form, like every object. Their size has to be set, which is shown with a black, jerky border then the displayed picture has to be loaded. This task is executable in the development time and in the run time, too.



Figure 8: TMediaPlayer serves the treatment of media elements

TMediaPlayer type (Figure 8) supports playing of audio and video data. The manager surface seen on the figure appears on our application which can start, stop and step the theme. The Display property contains the place of the appearance of the video.

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

3.3.2. Handling the frames of a video

The position property of TMediaPlayer type gives the number of the frame even if the video is not played. TPanel type may be the place of tracing out the video on the application. We can manage the clicking on this panel with the OnClick event of this TPanel and we can save the position of the mouse with the help of OnMouseDown event. With this knowledge we have all to create the necessary data registration program.

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

Figure 9: The data appear in a table with two columns

In this way we can save the data, the X and Y coordinates of the mouse positions into a table (Figure 9). The figures in parenthesis on the header of the table show the coordinates of the centre of the pendulum.

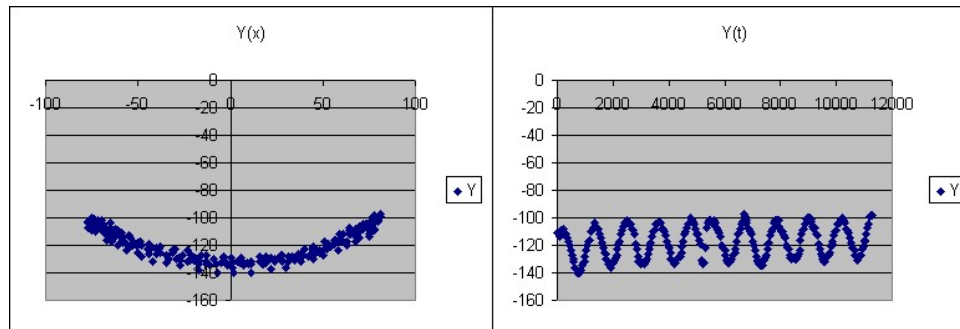


Figure 10: Data of the Y evasion in the function of X and the time (T)

These values can be saved into a text file in which data are separated with tab character. This file can be loaded in by Excel and the data can be evaluated (Figure 10).

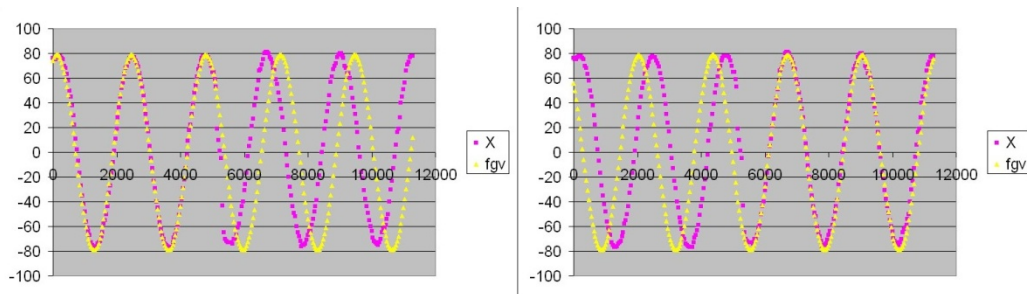


Figure 11: The graphs fitted onto the evasion, before and after the offset

The video contains a little jump in the older version, because the video was running during the digitalisation but some frames got lost while the slower computer was saving the buffered data.

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

The purple dots on the Figure 11 are the data from the video, the yellows ones are the values counted with (1) formula. We can count the difference from the two offsets (α), so 10 frames were left out. T was calculated as 2,33575662 seconds, so the mistake is 17%. It is quite considerable, but I think it is acceptable if we take the errors into consideration. There could be errors in a lot of steps. The older video was recorded by hand without a stage. The digitalisation can bring errors as we saw. We got data by mouse clicking, not necessarily hitting the centre of gravity. The values of the formula were set approximately so that in the Excel there is coverage on the graph.

3.3.3. More efficient data registration – tracing

But the previous human process is slow. If there is 10-second-long video, 250 clicks are needed. At first it can be good as I wrote in the previous paragraph, but in the long run this may slow down the work. That is why it would be practical to build in a feature into the application that can support the data registration. Defining the place of the object is possible in two ways. According to one of the approaches we should use the edges of the object, and the other one advises to manage the whole object in one. I selected the centre of gravity calculation because it is simpler in this raster picture. We do not need to get the edge of the object or calculate the centre of a possible concave object, but we can calculate the centre of any shapes.

My idea is that we should generate a black-and-white picture from the frames, where the bigger area is white and the target object is black. Then we calculate the centre of gravity of the picture that is we get the centre of the black dots. I assumed that the observed body is almost completely homogeneous and

centrally symmetric. This tracing can still be refined by calculating the little domain of the object instead of the whole picture. This domain must be the place where the object can move in 40 ms.

The next little trick helps to reach the frames as pictures.

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

```

If we use the code above, we may attain in our program the Canvas of the panel too, we can digest the pixels of the pictures on the panel. We can obtain the colour of the drawn picture (I,J) of a frame on its position with the next command.

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

At first we have to do filtration on the picture, then we have to calculate the centre of gravity of the picture. We digest the picture by pixels and we assign all pixels a 1 to the observed object and 0 to the others. The easiest way to get contrast pictures is to put a white object in front of dark background or vice versa. Now we can tell pixel by pixel from the RGB components if there is an object or not. The positions of the object were there on our video where the colour of the pixel is whiter than a given grey tone. This pixel will be drawn as a black dot and its value will be 1.

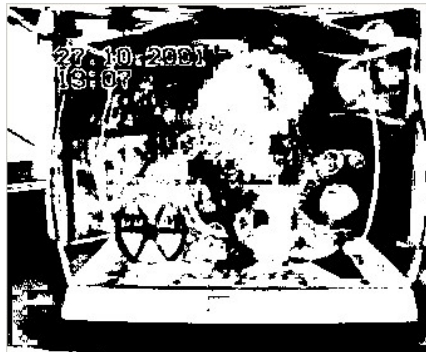


Figure 12: The result of black and white filtration

This filtration cannot be used yet on the first uptake (Figure 12), because we did not make sure that the object and the background are rather hard-contrast, but our laboratory uptake (Figure 13) already met this requirement.

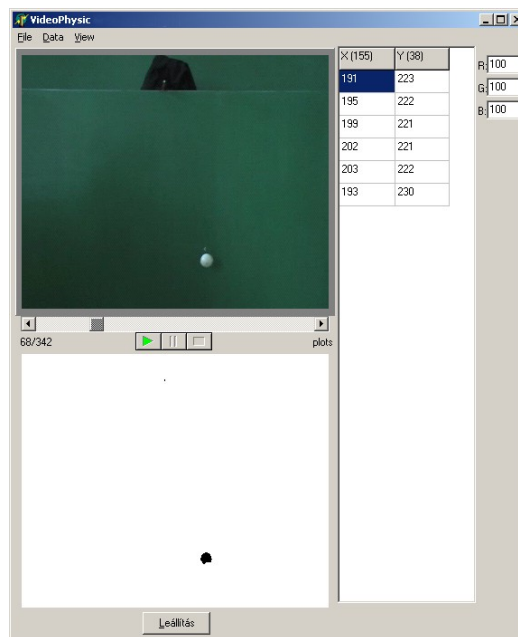


Figure 13: The program supports the tracing with filtration

4. Results

I know that there are still a lot of opportunities for development, I did not present an end-product, but an idea, which may be exploitable in the education with little creativity and cleverness. This is cheap, interesting and easily feasible with the use of the developing and current technologies. This article can be thought over and processed on a high-school subject interest group from the viewpoint of the presented interdisciplinary curricula, the source codes and programming technology. The videos, the source code and Excel table for evaluation can be reached at this link: <http://xml.inf.elte.hu/articles/MultimediaDelphi>

Bibliography

1. Új Pedagógiai Szemle; Komenczi Bertalan: Didaktika elektromagna? Az e-learning virtuális valóságai 2004. November <http://www.oki.hu/oldal.php?tipus=cikk&kod=2004-11-ta-Komenczi-Didaktika>
2. OKI; Radnóti Katalin: A fizika tantárgy helyzetét és fejlesztési feladatait feltáró tanulmány 2002 <http://www.oki.hu/oldal.php?tipus=cikk&kod=szakmai-2002-Radnoti>

Other Links

1. VideoPoint <http://www.lsw.com/videopoint/> and 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
4. Delphi <http://delphi.about.com/od/beginners/a/dbeginner6.htm> and <http://delphi.about.com/>

Author

László Menyhárt, Eötvös Loránt University, Budapest, Hungary. E-mail: menyhart@elte.hu