STIVID: A VBSCRIPT-based program for adding visual and auditory stimuli on videotapes

ARMAND DE CLERCQ, ANN BUYSSE, HERBERT ROEYERS, LESLEY VERHOFSTADT,
INGE ANTROP, KIM DE CORTE, and OLIVIER PEENE
Ghent University, Ghent, Belgium

Some psychological experiments require placement of visual and auditory stimuli on predefined frames in a videotape. We introduce STIVID (STImuli on VIDEO) as a method of performing this task. STIVID can add images, words, simple shapes, and audio tones to specific frames of existing video files created in AVI format. STIVID is written in Visual Basic and uses VBScripts to modify the AVI files. Three applications of this program are discussed: adding stimuli for reaction time experiments with couples watching a relational conflict, use of distractors in ADHD experiments, and placing eye masks on actors in existing videotapes to manipulate perceptibility of the target's facial expression during the mind-reading process.

In psychological experiments, placement of visual and auditory stimuli in predefined frames in a videotape is often required. This is not easily accomplished with standard video-editing software such as Adobe Premiere (Adobe, 2003). Therefore, we developed STIVID (STImuli on VIDEO) to add images, words, simple shapes, and audio tones to existing video files in specific frames. Examples of applications of this tool include experiments for measuring choice reaction times in a dual-task paradigm. Two subjects had to respond to visual stimuli by pressing a mouse key. The image targets were superimposed on a videotape and were as short as one frame. Short tones were placed on one stereo channel and were used by the computer as a time marker. The visual stimuli and time markers were synchronous, with millisecond accuracy. Placement of these stimuli could not be realized with standard video-editing software. In a second application, STIVID was used to investigate the effects of the temporal nature of stimulation during a waiting period. There were two waiting conditions, with the presentation of a row of squares and tones combined with the presentation of a silent movie. In one condition, the stimulation was temporally ordered, whereas in the other the stimulation was randomly ordered. In a third application, STIVID was used to create a continuously moving white square that masked the eyes or other parts of the face of people shown in a video. The software allows the independent placement of the square within each frame.

METHOD

General Information

Video editing with computers involves several steps. First, the original video is played back on a standard

VCR. The video output (images and sound) is fed into a computer through a video processing card. In most cases, the accompanying software is provided by the video card company. In most cases, video data is saved on PC in Windows AVI or Apple QuickTime format. In both formats, video data is stored on a number of tracks. One track contains the video image as a number of consecutive discrete images. Two other tracks are used for the stereo sound. Files in AVI as well as in QuickTime format can be edited and then played back on screen through standard software such as Windows Media player, or they can be sent back to a VCR through the same video card that was used to transfer the video to the PC. Both Apple and Microsoft offer software packages for editing and manipulating video and audio tracks on a frame-by-frame basis. For Apple (2003), this is the set of QuickTime API routines described in the Quicktime Developer documentation. This documentation includes conceptual material, code samples, and reference material (functions and data types). The conceptual material is intended for all QuickTime programmers. The reference material is aimed at procedural C/C++ developers. Microsoft (2003) offers the Video for Windows API as a part of the Windows SDK package. Since STIVID is intended for editing .AVI files, the Video for Windows API is used. The aim was to write all software in Visual Basic 6, because programs written in this language are much easier to modify and update. Although the Windows SDK package is intended for use in C++, it is possible to use the API functions from Visual Basic. All examples and documentation from Microsoft, however, are written for C++. This makes the use of the SDK package for Visual Basic programmers a nontrivial task. For this reason, STIVID was first developed in C++ and, after testing, completely rewritten in Visual Basic 6. Accessing the .AVI file by using the API routines offers the advantage that no constraints are imposed on the type of video compression (codec) that is used to create the video file. After STIVID is started, the program asks the user to select the

The authors thank the reviewers for their valuable comments on a previous draft of the manuscript. Correspondence concerning this article should be addressed to A. De Clercq, Department of Applied Mathematics and Computer Science, Ghent University, Krijgslaan 281 S9, B9000 Ghent, Belgium (e-mail: armand.declercq@rug.ac.be).

codec from a list of possible items that are installed in the computer.

Software Description

The program is created in Visual Basic 6 as a Windows COM object known as an ActiveX EXE file. This COM object can be used in a VBScript, as is the case in the first two applications, or it can be used interactively in another program, as is the case in the mask application. The objective is to allow any programmer who is familiar with Visual Basic to use the COM object for his or her specific requirements. The program runs under any version of Windows. It was tested under Windows98, Windows2000, and WindowsXP. There is no need to install the Visual Basic compiler on the PC, because the VBScript compiler is included in all Windows versions. STIVID uses a one-frame graphical buffer called the *canvas*, which is displayed on the right when the program is running, as is shown in Figure 1. For this canvas, a color can be defined that acts as a transparent color. Using this COM object in VBScript or in another program consists essentially of defining a series of stimulus blocks comprising a start-frame and an end-frame number and a series of commands that put graphical data on the canvas. For each frame between the first and the last frames of the stimulus block, every pixel is replaced by the corresponding pixel that does not have the transparent color in the canvas. The canvas can contain elementary figures, such as rectangles, images imported from .BMP files, or words. The transparency mode can be switched off for images from .BMP files. In this case, every pixel of the frame is replaced by the corresponding pixel in the .BMP file. Video frames for which no stimulus block is defined are simply copied from the source to the target AVI file. In every stimulus block, a sine wave tone can be defined, which is added to one or both sound tracks (left and right) and synchronized with the first frame of the stimulus block. A stimulus block can be as short as one frame. Thus, short stimuli—that

is, those the length of one video frame with accompanying audio beep—can be created. Note that this audio beep can be for either the computer data logger or the subject. A demo VBScript for using STIVID is presented in Listing 1 (see the Appendix). Each line is numbered and is followed by one or more comments. Each comment line starts with `'`, which is the syntax for inserting a comment in a VBScript program.

Modifying images and sounds in frames is done in stimulus blocks. Each stimulus block starts with a line such as 17 and ends with a line such as 20 in Listing 1. It is the responsibility of the user that all timing be given as an exact multiple of the timing of one frame. In a video, stimuli are displayed on a frame-by-frame basis. Defining a stimulus time that does not start at the beginning of a frame results in unpredictable timing errors. Therefore, STIVID checks this and generates an error if the user violates this rule.

Squares can be drawn instead of figures. In this case, Line 18 is replaced with a line such as the following:

```
VID.FigLine Color, Width, x1, y1, x2, y2,
BOX_FILLED
```

It is also possible to use words as stimuli. In this case, font, size, and the placement of the word on the screen are defined with a line such as the following:

```
VID.Text "RIGHT2",65535,"Courier
New",48,TEXT_CENTER +
TEXT_BASELINE,FrameWidth / 2,FrameHeight / 2
```

APPLICATIONS OF STIVID

STIVIDCouple

The first application, called STIVIDCouple, is used to measure the processing of relational information by romantic partners. It is a rather sophisticated program created for one particular experiment. In that experiment, both spouses had to respond to X or O stimuli that

Figure 1. STIVID uses a one-frame graphical buffer, called the *canvas*, which is displayed on the right when the program is running.

were superimposed on a videotape showing a relational conflict. The beginning of each trial was signaled by a warning stimulus consisting of two small asterisks (**), one frame in duration, that appeared at the center of the screen. This warning stimulus signaled the participants to press the center home button of the three-button LOG-ITECH mouse with the index finger of the dominant hand. The participants were further instructed to keep the finger on the home button and to press the left button with the index finger when the target stimulus was an X and the right button when the target stimulus was an O (see De Clercq & Buysse, 1999, for more details concerning this experiment). The X and O stimuli could be replaced by small figures or words that appeared in the center of the screen. In an introduction, the participants were told for which images or word the right and left mouse button needed to be pressed. The images were put on videotape. One stereo channel was reserved for the generation of a small beep that accompanied each image. The length of the beeps identified the warning, left, and right images. When the videotape was played back, the beeps were decoded by the computer that handled the mouse used by the participants. On this PC, called the *SlavePC*, an assembler program ran under MS-DOS. The PC was controlled by a second computer that ran under Windows, called the *MasterPC*, which checked and saved the data and was equipped with an easy-to-use graphical interface. It was impossible to use standard reaction time software running on a PC, such as INQUISIT from Millisecond Software or E-Prime (Psychology Software Tools, 2003; Schneider, 1998), because the stimuli had to be projected on the screen on which the video was shown. Therefore, the stimuli had to be placed on the videotape. We tested the synchronization between the start of the frame with the visual stimuli and the start of the sound beep when the video was played back on a VCR. A white square was used as the visual stimulus. A black square was placed in all the other frames. The start of the stimulus was detected by a photocell placed in front of the computer display at the place where the squares occurred. The rise time of the photocell was 10 μ sec, and the fall time was 200 μ sec. The rise and fall times included the values for the photocell with accompanying amplifier and the persistence of the screen phosphor. The rise time of the photocell was sufficient for detection of visual information with millisecond accuracy (De Clercq, Crombez, Buysse, & Roeyens, in press). The output of this photocell was connected to a pin of the parallel printer port of a PC. The output of one sound channel of the VCR was amplified with a small amplifier in such a way that peaks of the sound sine waves could be detected as 0/1 signal when it was connected to another pin of the parallel printer port of the same PC. The appearance of the first peak of the sine wave was then compared with the appearance of the photocell signal. On this PC (the *SlavePC*), an assembler program ran under DOS. This PC was controlled by a second computer that ran FASTLOG (De Clercq et al., in press). Configuring the *SlavePC* under DOS ensured a timing

accuracy of this test system of 0.01 msec. It is impossible to obtain this accuracy in a DOS console running under Windows. For a standard VCR with an interlaced frame rate of 50 Hz (PAL), the synchronicity between visual stimulus and beep is accurate within 1 msec (see Table 1). When STIVIDCouple was started, a data grid was shown to the user. The user could select one or more files containing references to the left or right image or containing the words that had to be used in the experiment. Figure 2 shows the part of the data grid on which all the necessary parameters, such as frames per second, start time of the first warning on tape, minimum and maximum times between the warning and the left/right stimulus, minimum and maximum times between a stimulus and the next warning, and so on, had a predefined value. Each value, however, could be modified in the corresponding cell in the grid by clicking on it. After all the parameters were verified, a sequence of warning images and stimuli could be generated at random intervals. The result is shown in the next four columns of the table. In the last stage, the corresponding VBScript file, which will be used by STIVID, will be generated.

STIVIDADHD

The second application, STIVIDADHD, for creating distractors, is a rather simple program. All necessary information for generating the squares on videotape, such as color, size, place, accompanying sound, and mode of displaying the squares (random or in sequence), is placed as a comment line in a text file that is used to generate the VBScript file for using STIVID. In this way, the user can easily adjust the parameters without recompiling the program.

The Appendix shows an example of this file.

This file has three parts. The first part starts with <PART=PARAMETER> and contains the definitions of the parameters that are used in STIVIDADHD. The second part starts with <PART=PRE> and contains the start of the VBScript file that will be generated. The last part starts with <PART=POST> and is the terminating part for the VBScript file. Every parameter contains a short description. Using one file that contains all the necessary information has a number of advantages. First, all the information is kept together and is the only data that must be saved for further use. Second, the user can easily change the necessary script lines for input, output, and so on. In a number of applications, it is shown that end users find this way of using one text file to tailor an application as easy as using a sophisticated user interface with Windows.

Table 1
Timing Accuracy (Mean, Standard Deviation, and Maximum Deviation) Between Image and Sound Stimuli Created With STIVID on Standard VCRs During 20-Min Playback

Apparatus	Number of Samples	<i>M</i>	<i>SD</i>	Maximum Deviation
JVC SVHS HR-S7500E/EH	108	14.3	0.01	± 0.35
JVC VHS HR-J658EH	118	14.9	0.01	± 0.4

Note—Mean, *SD*, and maximum deviation are measured in milliseconds. Both VCRs are PAL systems with an interlaced frame rate of 50 Hz.

ParameterName	Param
FramesPerSecond	25
Time of first warning (msec)	10,000
Minimum time between Warning and Stimulus (msec)	640
Maximum time between Warning and Stimulus (msec)	2,640
Minimum time between Stimulus and next Warning (msec)	3,000
Maximum time between Stimulus and next Warning (msec)	5,000
Duration of Warning in Frames(F) or msec(M)	F1
Duration of L/R stimuli Frames(F) or msec(M)	M1600
Total Number of stimuli	82
TapeNumber	901
WarningImage	I:\Proje
CollectionSize	82
Protection	ON
TextFont	Courier
Text Size in points	48
Text Color	65535
TapeUserMessage	EXP 20

Figure 2. Part of the main screen from the STIVIDCouple application showing the parameters that can be modified.

STIVIDMASK

STIVID can also be used directly in another program. This option is used in the third application, called STIVIDMask. In STIVIDMask, the user can create a white square that masks the eyes or other parts of the face of subjects in a videotape. The videotape is played back in a window. The playback speed is controlled by the user by pressing keyboard keys. At the same time, the user can move the mask with the mouse. In this case,

STIVIDMask starts STIVID with the information necessary to open the .AVI file on screen (this is shown in the left window in Figure 3). Because STIVID is an ActiveX EXE file, it is possible for another program to communicate with it by calling the COM objects in it. In fact, this is the same procedure that is used in the VBScript files in the preceding two applications. Then, the STIVIDMask program can call STIVID to display the requested video frames and can call STIVID with a FigLine method to draw the mask square directly on screen. In this way, the user can see directly where the mask will be placed on each frame. In this interactive mode, however, no modifications are made to the actual .AVI file. For this reason, STIVIDMask does not open the VID object with the line

```
Set VID = CreateObject("RUG_STIVID.VID")
```

but with

```
Set VID = CreateObject("RUG_STIVID.VIDRO")
```

The .AVI file is opened in read-only mode. The mask program keeps a record of the placement of the mask in each frame. At the end, a VBScript file, such as that explained above, is created to do the necessary modifications.

DISCUSSION

Video manipulation has a long tradition in psychological experiments. Video editing is used for self modeling (Dowrick & Raeburn, 1977, 1995), various fields of the behavioral sciences (Dowrick, 1991), visual perception (Unuma, 1992), and other domains (Morgan, 1998). In those applications, parts of the video are removed, moved,

Figure 3. STIVIDMask (right window) communicates directly with STIVID (left window).

or modified using standard video-editing techniques. Editing can be done using dedicated hardware or software. There exist a number of professional video-editing software programs, such as Adobe Premiere (Adobe, 2003), that can be used for such applications. They are not intended for use in dedicated psychological applications such as those explained here. Adobe Premiere permits the user to write special plug-ins that can do some of these tasks. Those plug-ins must be written in the C language and, therefore, require skillful programming experience. Moreover, expensive commercial software is needed. STIVID is written in Visual Basic and uses VBScripts. This requires minimal knowledge of the VBScript language. Any psychologist with basic programming knowledge can create it. Furthermore, STIVID is freeware; the complete source can be obtained from the first author. STIVID can be used in a number of areas, Visual Speech research perhaps being one of them.

We have discussed three research areas in which STIVID is used in our laboratory. Each of the three applications discussed shows a specific area and method for using STIVID. The third application, for creating eye masks, shows how STIVID can be used interactively. It is an example of an application that is normally available only in dedicated and expensive professional video-editing software. The software is free and can be downloaded from <http://allserv.rug.ac.be/~adeclerc/stivid>. The website includes a simple demo program that shows how to use STIVID and contains detailed information concerning the installation process. All sources of STIVID software will be made available on the Internet after publication of this article.

REFERENCES

- ADOBE SYSTEMS, INC. (2003). Adobe Premiere Information [on line]. Retrieved February 3, 2003, from <http://www.adobe.com/products/premiere/main.html>
- APPLE COMPUTER, INC. (2003). Quicktime Developer Documentation [on line]. Retrieved February 3, 2003, from <http://developer.apple.com/techpubs/quicktime/quicktime.html>
- DE CLERCQ, A., & BUYSE, A. (1999). An inexpensive system for measuring the processing of relational information by romantic partners. *Behavior Research Methods, Instruments, & Computers*, **31**, 299-304.
- DE CLERCQ, A., CROMBEZ, G., BUYSE, A., & ROEYERS, H. (in press). A simple and sensitive method to measure timing accuracy. *Behavior Research Methods, Instruments, & Computers*.
- DOWRICK, P. W. (1991). *Practical guide to using video in the behavioral sciences*. New York: Wiley.
- DOWRICK, P. W., & RAEBURN, J. M. (1977). Video editing and medication to produce a therapeutic self-model. *Journal of Consulting & Clinical Psychology*, **45**, 1156-1158.
- DOWRICK, P. W., & RAEBURN, J. M. (1995). Self-modeling: Rapid skill training for children with physical disabilities. *Journal of Developmental & Physical Disabilities*, **7**, 25-37.
- MICROSOFT CORP. (2003). Platform SDK: Windows Multimedia: Video for Windows [on line]. Retrieved February 3, 2003, from http://msdn.microsoft.com/library/default.asp?url=/library/en-us/multimed/avifile_8dgz.asp
- MORGAN, R. E. (1998). The perception of body structure, symmetry, and movement by 3- to 5-month-old infants. *Dissertation Abstracts International: B. The Sciences and Engineering*, **59**, 1887.
- PSYCHOLOGY SOFTWARE TOOLS (2003). E-Prime. Retrieved February 3, 2003, from <http://www.pstnet.com/e-prime/default.html>
- SCHNEIDER, W. (1998, November). *E-Prime: A cross platform experiment generator studio for computerized behavioral research*. Paper presented at the Commercial Symposium of the 28th Annual Conference of the Society for Computers in Psychology, Dallas, TX.
- UNUMA, H. (1992). Spatio-temporal integration in visual perception: Sequential part presentation and two levels of information integration. *Japanese Psychological Research*, **34**, 158-164.

APPENDIX

Listing 1

```

Const SelectRight = 1, SelectLeft = 2                                1
const NOclearChannel = 0, ClearChannel = 1                          2
Const TEXT_LEFT = 0, TEXT_RIGHT = 2, TEXT_CENTER = 6, TEXT_TOP = 0  3
Const TEXT_BOTTOM = 8, TEXT_BASELINE = 24                          4
Const TEXT_BOLD = &H100, TEXT_ITALIC = &H200                       5
Const BOX_HOLLOW = 1, BOX_FILLED = 2                               6
Const FIG_LEFT = 0, FIG_RIGHT = 2, FIG_HCENTER = 6, FIG_TOP = 0    7
Const FIG_BOTTOM = 8, FIG_VCENTER = 24                             8
Dim VID, FrameHeight, FrameWidth, FrameTime                        9
' All constants and variables are defined at the beginning of the
' script with Lines 1 to 9
Set VID = CreateObject("RUG_STIVID.VID")                            10
' An instance of the STIVID object is created here
VID.AVIOpen "filein.avi","fileout.avi","log.txt"                    11
' Filenames of input, output and log file are defined in Line 11
VID.SoundInit SelectLeft, ClearChannel                              12
' Line 12 selects the audio channel that will be used
VID.BackgroundColor 0                                              13
' line 13 selects the transparency color for the stimuli
FrameHeight = VID.FrameHeight                                       14
FrameWidth = VID.FrameWidth                                         15
' Lines 14 and 15 return the width and height of each video frame.
' These 2 parameters can be used to place drawings on the canvas whose
position is related to the size of the image
FrameTime = VID.FrameTime                                           16
' This line returns the length of one frame in milliseconds
VID.StartBlock "MA",30000                                           17
' Start of a stimulus block. "MA" tells the system that the next
' argument is the start time in the .AVI file in milliseconds.
VID.BMP "stimulus.bmp",FIG_HCENTER+FIG_VCENTER,Width,Height        18
' This line describes the .BMP file that will be put on all the frames
' of the stimulus block. The extra arguments indicate the
' place and size of the image in the frame.
VID.Beep 250,32,100                                                 19
' This line indicates the sound (frequency, duration, and amplitude)
' that will be started at the beginning of the block.
VID.EndBlock "FS",1                                                 20
' The stimulus block ends with this line. "FS" indicates that the next
' argument is the number of frames in which the stimuli painted on the
' canvas will be used. Other combinations for indicating the start and
' the stop time of this block are possible.
...
VID.CloseAll                                                         21
Set VID = Nothing                                                    22
' Lines 21 and 22 terminate STIVID.

```

An Example File of STIVIDADHD

```

<PART=PARAMETER> PARAMETERS SECTION. All parameters are between <>
<CONDITION=1> =1 all blocks in sequence, =2 random
' in sequence, colors and sound volumes are sequentially used
' from color and volume array (see below)
<CYCLES=1> number of cycles for which one sequence is repeated
' original (1,020/(3*15))=300 if BLOCKSPERTRIAL=15
<BLOCKSPERTRIAL=15>
<LEFTSCREENOFFSET=30> All screen data is on 1/1,000 of full screen
<RIGHTSCREENOFFSET=92>
<BLOCKHEIGHT=92> Y value of block on screen 0.092 full screen
<BLOCKWIDTH=92> width of block 80+15%
<COLOR1=255,0,0>

```

APPENDIX (Continued)

```

' COLOR1 .. COLORN : array of RGB colors for each block
<VOLUME1=100>
' VOLUME1 .. VOLUMEN : array of sound volumes for each block
<TONEFREQUENCY=250> in Hertz
<TONELENGTH=300> tone length in milliseconds
<BASETIMEBETWEENIMAGES=75> in FRAMES
<IMAGEDURATION=8> in FRAMES !!!
<TIMEDEVIATION=0>
' maximum number of frames by which the time between TWO frames can deviate
' for sequential: this is 0
<PART=PRE> HEADER VBS part
Const SelectRight = 1, SelectLeft = 2, NOClearChannel = 0, ClearChannel = 1
Const TEXT_LEFT = 0, TEXT_RIGHT = 2, TEXT_CENTER = 6, TEXT_TOP = 0,
TEXT_BOTTOM = 8, TEXT_BASELINE = 24
Const TEXT_BOLD = &H100, TEXT_ITALIC = &H200
Const BOX_HOLLOW = 1, BOX_FILLED = 2
Const FIG_LEFT = 0, FIG_RIGHT = 2, FIG_HCENTER = 6, FIG_TOP = 0
const FIG_BOTTOM = 8, FIG_VCENTER = 24
Dim VID
Dim FrameHeight, FrameWidth, FrameTime
Set VID = CreateObject("RUG_STIVID.VID")
' in next line the first argument must be the .AVI file to read,
' the second argument the output file, and the third argument the log file
VID.AVIOpen "fin.avi";"d:\tout.avi";"c:\temp\info.log"
VID.SoundInit SelectLeft+SelectRight , NOClearChannel
VID.BackgroundColor 0
VID.PercentValues=true
<PART=POST>
VID.CloseAll
Set VID = Nothing

```

(Manuscript received November 12, 2002;
accepted for publication February 12, 2003.)