Personal WebMedia for Windows (PWM)

WebMedia for Windows was written by Travis Heppe, Rainer Birkenmeier and Heiner Wolf
in cooperation with Konrad Froitzheim, Holger Boenisch and Stephan Schmid.

Introduction.

PWM combines the following functionality into one package:

frame grabber/digitizer - PWM takes frames from a video capture device and saves them internally.

web video server - serves HTTP requests with streams of pictures one after another in a format understandable to Netscape.

web audio server - provides continuous audio streams of GSM or ADPCM format. The streams are requested by clients using a protocol similar to the well-known RTSP.

micro web server - a minimal web server implementation which allows transmission of HTML files, Java applets, and static pictures.

stream server - PWM accepts video streams from outside through sockets and can serve them to HTTP requests exactly as if the pictures had come from the internal card.

stream converter - converts freely between the following image formats: raw24, JPEG and difference GIF

Chapter 1: **Running Personal WebMedia for Windows**

1.1 Requirements Summary

- a PC with Windows 95/NT (we recommend Pentium 100 or higher with 32 MB)
- a video grabber card or QuickCam
- a Soundblaster 16 compatible sound card
- Soundblaster driver version 4.30.0/4.0.4 or higher (Windows 95/WindowsNT)

1.2 Running Video with PWM

For PWM to run properly, a video digitizer is necessary. The system has been tested with the cards listed below, but there is no reason why other Video for Windows (VfW, the video subsystem for Windows 95/NT) compliant devices could not be used, with or without hardware JPEG compression.

PWM compatibility list:

- Miro DC1, DC20, DC30
- Fast Movie Machine II
- Creative Labs Video Blaster RT300
- QuickCam

The next item of business is an appropriate driver for the card. The driver needs to be something recognised by Video for Windows as a "video capture driver". If multiple drivers are installed, Video for Windows will provide a list of drivers in alphabetical order, and PWM will attempt to connect to the first.

The video options are stored in a configuration file called "PWM.cfg". The file doesn't have to be present, it will be generated automatically during application shut down if it doesn't exist. Thereafter it holds the default values. Lines denoting options contain one keyword-value pair of the form "<option_name> = <value>" per line.

1.3 Running Audio with PWM

To provide audio data from the local machine on the Internet, one needs to enable audio from the appropriate menu. The audio server listens on a designated port (default port 7777) for stream requests. A protocol, similar to the well-known RTSP protocol, is used by the server and client to set up an audio stream.

Multiple clients may connect to one single audio server which delivers multiple audio streams to all receivers. The audio client is a Windows 95 plug-in for Netscape Navigator 3.0 or above (see Installation Notes). The plug-in starts automatically while browsing on Web pages containing an embed tag which specifies the audio stream. Supported audio formats for communication are GSM 6.10 and ADPCM. GSM is highly optimized for speech coding while ADPCM is a more general purpose audio format.

In order to listen simultaneously to multiple audio streams, a wave mixing library is provided. All running plug-ins (one for each receiving stream) write the receiving audio data in the shared

memory of the mixing library. The library mixes the various streams and outputs the wave data on the sound device.

Audio and video share the same configuration file.

Chapter 2: **The user interface**

There are two options for controlling **WebMedia**: locally through the graphical user interface or remotely through telnet. The full flexibility of **WebMedia**, however, could only be provided locally through the graphical user interface.

2.1 The File-Menu

Save Picture
You can save the current frame as JPEG-picture if you enter a name for it in this dialog-box and press "OK". The given name can include a complete path. It is relative to the directory you started **WebMedia** from, unless the path starts with a backslash and/or a drive letter.

Exit
This menu item exits **WebMedia for Windows**. All settings are stored in the configuration file "pwm.cfg", so the next time you start **WebMedia** you will find it in the same state you left it.

2.2 The Video-Menu

Video Format
The settings that are possible here depend on the kind of frame grabber and especially on the drivers you use with Video for Windows. If any problems occur here, please consult the documentation of your frame grabber for detailed information.

Take care not to choose the 16-bit RGB format. This format is not supported by PWM and can cause unpredictable results!

Video Source
As the "Video Format"-menu this depends on your hardware/driver combination, consult the documentation for help.

Local Display
The sub-menu of this item shows whether PWM displays every new picture it gets from the video hardware or not. It's much nicer to see the whole video that is spread into the world but, depending on your configuration, this might cause heavy system load. If you disable this preview, the local display slows down to about one picture every 5 seconds. It is not disabled completely to give you a hint about what the camera actually shows, this prevents you from drifting out of the frame in longer sessions.

Video Setting
Source Tab: PWM can process input from either an internal frame grabber or an external source. The internal frame grabber is specified by the device number. The devices are enumerated starting with 0. This option is very useful if you e.g. use a QuickCam in addition to your standard frame grabber. If there is no corresponding device to the number you type in the output window

will remain black. The interface to an external source is specified by a port address (default port is 10000).

Speed Tab: The frame rate that you can supply depends on the speed of your computer system. You can adjust the frame rate of the frame grabber card manually using a slider which specifies the delay between to frames. It covers a range from 50ms (=20 Frames per Seconds) to 5000ms (a frame every 5 seconds). The slider is called "Average Framerate" because you may set the related parameter "Dynamic Range" too. If you drag the dynamic range trackbar to its leftmost position the frame rate is fixed at the average delay parameter. The more you drag the trackbar to the right the more the real frame rate varies. It varies depending on the current processor load. If your system is heavy-loaded the frame rate will decrease and vice versa.

JPEG Tab: Since PWM is independent of the underlying hardware it has a built-in JPEG-compressor. If your frame grabber has no JPEG-compression on board, conversion is done automatically by PWM. In this dialog-box you can change the level of quality for this conversion, which affects the data volume that has to be transferred to the client (if it uses the JPEG-format). Don't be confused if even reducing the quality to zero has no effect on your local display. To avoid useless computations the JPEG-picture is built only if required and only for the outgoing video stream to the client.

GIF Tab: These setting affect the behaviour of the difference GIF algorithm, which distributes only those parts of the picture that have changed. Here the sensitivity of the difference detection can be tuned by the tolerance. As a tuning aid small white borders can be drawn around the rectangles while transmitting them if you check "Sub-Picture Frames Visible" (Note that they are only visible on the client side).

Tolerance for red, green and blue:

Classification between noise and real change is done by color. With this option you can specify the threshold value for each color component.

Block parameters:

The min size specifies the noise detection block size. The x-/y-distance parameters control the generation of GIF-subimages. They specify the minimum distance of separate subimages.

2.3 The Audio-Menu

Enable Audio

Switches the audio part of PWM on/off completely. All connections are affected. If you want to mute a single client only use the appropriate button of the connection window (see option "Show Connections").

Audio Settings

Quality of Service Tab: If you want a report on audio quality PWM can generate a (QoS) report. This report contains information on packet delay, number of dropouts and jitter. Obviously you may alter the audio port too.

Default Client Settings Tab: The data of the quality of service report is provided by the

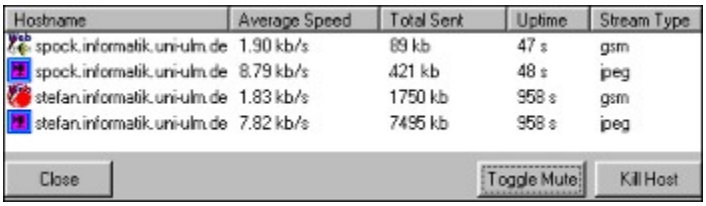
WebAudio plug-in. Hence, you can choose the type of the report as well as the format of the requested audio data (GSM or ADPCM) here. The buffersize parameter specifies the number of packets to be buffered on the client side to compensate packet delay. If your are talking to somebody the buffer should be as low as possible to yield real time communication. If you are just listening radio or if the network connection is bad you should increase this value.

2.4 The Options-Menu

Show Connections

To see who actually watches your video or listens to your audio this menu item opens a new window that shows the hostnames of the connected clients (see figure ???) and some further information. You can mute single audio streams by selecting one and pressing the "Toggle Mute" button. Muted audio streams are marked by an icon which is crossed out. If the audio client signals significant delays or frequent dropouts then the bad connection is visualized by a red point on the icon (see figure ???).

Furthermore you can kill any stream after selecting a host entry in the list by pressing the button "Kill Host". Alternatively you can access these commands by the menu bar of the window (not shown here).



The "Connected Hosts" window

Options

Webserver tab: The built-in web server has very little functionality so there are just two options which can be configured. First of all you can specify the HTTP-port. The "Root Path" entry contains the root-directory of the web server tree. This is the path that is added to all HTTP-requests handled by the micro web server. The path is relative to the actual directory unless it starts with a backslash and/or a drive-letter. You can also disable the web server functionality completely if you mark the "Disable Micro-Webserver" checkbox.

Connections: A really neat feature of PWM is the ability to decide whether an incoming request for a video stream should be accepted or not or if the user should be notified of an incoming connection. If you are having a video-conference for example, you can deny people who don't belong to the conference participants, to take part. If the dialog-box opens, the hostname of the requesting client is displayed, so you can decide if the client may receive the stream. To keep things clear the acknowledge-dialog is triggered only for audio or video requests, normal HTTP-connections are ignored by it. If you don't want to be bothered this dialog-box any more just uncheck the appropriate checkbox and the acknowledge feature will be disabled. If the

notification is enabled then the specified sound file is played on an incoming connection. Al last you can switch of the DNS lookup on incoming connection e.g. if you don't have a name server available.

Control tab: The remote control support of PWM enables the modification of many options via a TCP-connection on the Control Port, in the simplest case this can be done using telnet. The following functions are supported: help, exit, info, port, framedelay, lastsockerr. PWM for Windows has the ability to log the client activities. The log data is written to a file you can specify here. As for "Save Picture" the given path gets absolute with a leading backslash and/or drive letter. The file holds information about the time and date a client connected to and disconnected from your video or audio stream.

2.5 The Help-Menu

It shows this help.

Chapter 3: **The Difference Detection**

How does the difference detection work?

1. Two successive pictures are subtracted from each other. The result is a so called difference picture.
2. Then square areas of the difference picture sized by a given edge length are treated. Within each area is averaged by adding up the color components of each pixels separately and the result is divided by the number of pixels. Together these average color components represent the new color for this area.
3. With this new color (I call it difference color) the area is classified by the given parameters into movement or not movement (noise).
4. Depending on the classification the whole square is marked in a mask as movement or not movement.

So the main principle is classification by color. This means there are regions of color which indicate movement and other regions which indicate not movement, but merely noise. The difference picture that results from the subtraction of two successive pictures with no differences at all is a picture where all pixels are colored black. Therefore black is the color of "no movement". If two pictures are different (for example a traffic light in action) the changes of the color components are significant in the changed areas and the difference color has a quite high red, green or blue value. Thus colors with a high red, green or blue component indicate movement.

Comparing two consecutive pictures of a CCD camera, gives small differences even in areas showing no visible movement at all (noise). The absolute value of CCD noise is obviously small, because it does not show an effect of movement to the human eye. As mentioned above, the color of "no movement" is black. So, if the sample value of the camera changes a little bit the result in the difference picture is close to black. Hence the conclusion is, that everything in the difference picture that is black or is close to black does not show any movement. The question is, what close to means and which geometry this area within the RGB color cube has. On the other hand is it necessary to know the area which covers the colors that show movement. Potentially there is an area which is covered by both the "movement" and the "noise color area". That means a reliable detection of movement within a sequence of pictures in the described way is impossible. Separating the areas of movement and the areas of noise is the main problem. The best way would be to classify each single color value into the two groups. If for each color value one parameter is used, you get 16,7 million parameters in a common system with 24 bit color depth! The disadvantage of this method is obviously the huge effort the classification needs and the difficult adaptation to different sources. Therefore a geometric classification of the areas is now discussed. The aim is to separate the areas by mathematical equations.

One possibility is the distance to black. One can specify a limit and all pixels with a smaller distance to black are classified as noise and the others are classified as movement.

For this reason a simple algorithm was implemented, which cuts an (adjustable) small block in the black corner of the RGB color cube. (This is shown in figure 1)

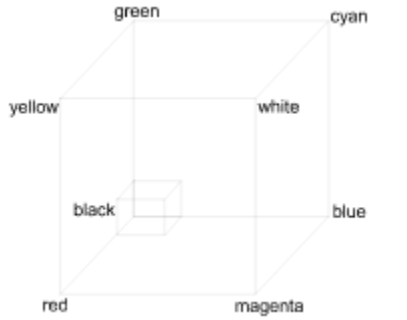


figure 1

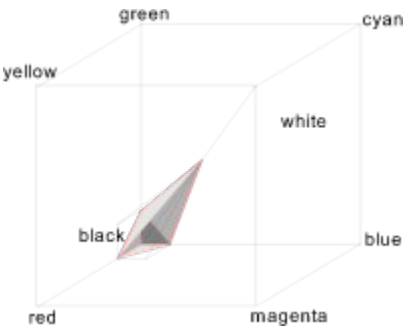


figure 2

The noise of different sources covers different areas. Viewing these areas shows that size and shaping is very dependent on the source and the preprocessing of the data.

Inside the investigated system the noise originated from a CCD camera and the JPEG compression.

The created noise is not easy to characterise geometrically. It looks like a pyramid standing in the black corner of the RGB color cube. The pyramid's top points to the white corner along the diagonal line connecting the black and the white corner (see figure 2). The shape of this area is a result of the colorspace conversion (RGB->YUV) used by the JPEG compression. The small noise of the sampling CCD in the camera is modified by the losses of the JPEG compression.

Detailed description for the values from the "Difference Settings" dialog box (it can be reached by selecting "Difference Settings" from the Options-menu):

Tolerance for red, green and blue:
Classification between noise and movement is done by color. With this option you can specify the maximum value for each color component. If the value of the component is below one of the given limits, the area is classified as noise. If they are all above, it's classified as movement.
[Graphic: Sidelength of the block in R,G and B direction. With -nsa: expansion of the pyramid at the points of intersection with the R, G and B axis]

Chapter 4: **Remote control through Telnet**

Though all commands are only available through the graphical user interface, there is also a text-based interface available through telnet. The following functions are supported: help, exit, info, port, framedelay, lastsockerr.

Chapter 5: **Pitfalls and known problems**

Instability: Miro DC1/DC20 requires a relatively large amount of memory for a DMA buffer. If there are too many active applications, a warning comes from Miro, and vid is unable to connect with the video camera. The only known solution to this problem is a reboot.

Dead connections: If the client doesn't close its connection gracefully the TCP connection will occasionally not be recognised as a dead one. Consequence the connection stays alive but no further data is sent. You have to kill this connection manually from the connection window.