Subject: Re: Writing mpeg from IDL
Posted by Robert.M.Candey on Tue, 12 Nov 1996 08:00:00 GMT
View Forum Message <> Reply to Message

In article <esmith-1211961607150001@news.gsfc.nasa.gov>,
esmith@hubble.gsfc.nasa.gov (Eric P. Smith) wrote:

> Several weeks ago someone posted an IDL procedure to write MPEG movie
> files which I saved and attempted to run. The output files of this
> procedure are not recognized by the UNIX mpeg_encode program which may be
> related to the fact that I do not have the subroutine PSEUDO_TO_TRUE.
> Unfortunately, in saving the file I truncated the poster's name and cannot
> contact him/her directly. Does anyone have the full post or the
> subroutine PSEUDO_TO_TRUE? Thanks in advance.
>
> --
> Eric P. Smith Laboratory for Astronomy & Solar Physics
> Code 681 esmith@hubble.gsfc.nasa.gov
> NASA/GSFC ericsmith@uit.gsfc.nasa.gov
> Greenbelt,MD 20771 (301)-286-8549

Eric, here is the post. Let me know how it works since I want to something
similar. Bobby

From: scott@abyss.ATMOS.ColoState.Edu (Scott Denning)
Newsgroups: comp.lang.idl-pvwave
Subject: Re: MPEG creation with IDL
Date: 18 Jan 1996 21:45:16 GMT
Organization: Colorado State University, Fort Collins, CO 80523
Lines: 90
Message-ID: <4dmf1c$1j9i@yuma.ACNS.ColoState.EDU>
References: <JACOBSEN.96Jan18065620@xray1.physics.sunysb.edu>
Reply-To: scott@abyss.ATMOS.ColoState.Edu (Scott Denning)
NNTP-Posting-Host: abyss.atmos.colostate.edu

The following idl procedure will produce an mpeg file from a series of
images stored in a 3D array (width x height x # of frames). It requires
the "mpeg_encode" executable to be inthe unix search path. This can be
obtained from ftp://s2k-ftp.cs.berkeley.edu/pub/multimedia/mpeg/encode.

There are lots of options that can be handled differently, to make
tradeoffs between image quality, speed, and disk space. See the
documentation for mpeg_encode for more details.

If your animation is stored in the array "image_array" and you want to
write it to a file called "movie.mpg," you would do so by typing

WRITE_MPEG, 'movie.mpg', image_array

Best of luck!

--
A. Scott Denning scott@abyss.Atmos.ColoState.edu
Dept. of Atmospheric Science Phone (970)491-2134
Colorado State University Fax1 (970)491-8428
Fort Collins, CO 80523-1371 Fax2 (970)491-8449

========== C U T H E R E ==========

```
PRO WRITE_MPEG, mpegFileName, image_array

movieSize = SIZE(image_array)
xSize = movieSize(1)
ySize = movieSize(2)
nFrames = movieSize(3)

nDigits = 1+FIX(ALOG10(nFrames))
formatString = STRCOMPRESS('(i'+STRING(nDigits)+'.'+STRING(nDigits)$
           +           ')', /REMOVE_ALL)
; Load current color table into byte arrays
TVLCT, red, green, blue, /GET
red = BYTE(red)
green = BYTE(green)
blue = BYTE(blue)

ON_IOERROR, badWrite

; Make a temporary directory if necessary or clear it otherwise'
TMPDIR = '/tmp/idl2mpeg.frames'
SPAWN, 'if (-d ' + TMPDIR + ') echo "exists"', result
dirExists = result(0) EQ 'exists'
IF dirExists THEN command = 'rm ' + TMPDIR + '/*' $
  ELSE command = 'mkdir ' + TMPDIR
SPAWN, command

; Write each frame into TMPDIR as an 8-bit .gif image file
FOR frameNum = 0, nFrames-1 DO BEGIN
  fileName = TMPDIR + '/frame.' + STRING(frameNum,FORMAT=formatString)$
         + '.gif'
  WRITE_GIF, fileName, image_array(*,*,frameNum), red, green, blue
ENDFOR

; Build the mpeg parameter file
paramFile = TMPDIR + '/idl2mpeg.params'
OPENW, unit, paramFile, /GET_LUN
PRINTF, unit, 'PATTERN         IBBBBBBBBBBP'
```

```
PRINTF, unit, 'OUTPUT          ' + mpegFileName
PRINTF, unit, 'GOP_SIZE 12'
PRINTF, unit, 'SLICES_PER_FRAME 5'
PRINTF, unit, 'BASE_FILE_FORMAT PPM'
PRINTF, unit, 'INPUT_CONVERT    giftoppm *'
PRINTF, unit, 'INPUT_DIR        /tmp/idl2mpeg.frames'
PRINTF, unit, 'INPUT'
PRINTF, unit, '`ls *.gif`'
PRINTF, unit, 'END_INPUT'
PRINTF, unit, 'PIXEL           FULL'
PRINTF, unit, 'RANGE            5'
PRINTF, unit, 'PSEARCH_ALG      LOGARITHMIC'
PRINTF, unit, 'BSEARCH_ALG      SIMPLE'
PRINTF, unit, 'IQSCALE          8'
PRINTF, unit, 'PQSCALE          8'
PRINTF, unit, 'BQSCALE          8'
PRINTF, unit, 'REFERENCE_FRAME  DECODED'
PRINTF, unit, 'FORCE_ENCODE_LAST_FRAME'
FREE_LUN, unit

; spawn a shell to process the mpeg_encode command
SPAWN, 'mpeg_encode ' + paramFile

RETURN

badWrite:
alert, 'Unable to write MPEG file!'

END
```