Software Development Seminar

Animation (Advanced)

New Features of the
Material Editor

Topics

New features starting with Ver2.0 -

- Selection of polygons in a region
- Control of motion and rotation speeds
  for objects
- Quadrangulation
- Material attenuation
- Automatic placement of TIM
- Specifying texture regions
- Fine adjustment of UV
- Compatible with Windows95

Selection of polygons in a region

1) Drag the mouse to define a rectangular region

2) All polygons within the region will be selected

Drag: select polygons in region (front)
Ctrl + drag: deselect polygons in region (front)
Shift + drag: select polygons in region (front and back)
Ctrl + Shift + drag: deselect polygons in region (front and back)

* The conventional method (selection by clicking) can also be used.

Control of motion and rotation speeds of objects

An object's speed of rotation and translation can be specified according to four levels:

Shift + 1: Translation 1
Shift + 2: Translation 4
Shift + 3: Translation 32
Shift + 4: Translation 100 (default)

Ctrl + 1: Rotation speed 1
Ctrl + 2: Rotation speed 4
Ctrl + 3: Rotation speed 8
Ctrl + 4: Rotation speed 16 (default)

Quadrangulation

Two three-sided polygons can be combined into one four-sided polygon

Material

RSD File : [   ]
Polygons : [   ]

Polygon ID : [   ]

Shading
- ○ Flat   ○ Gouraud

Textures
- ○ Texture ○ Color

Light Calc
- ○ on   ○ off

Display
- ○ on   ○ off

○ 50(B)+50(F)   ○ 100(B)+100(F)
○ 100(B)-100(F) ○ 100(B)+25(F)
○ 0(B)+100(F)   opaque

○ 1   ○ 2   ○ 3   ○ 4

[copy]   [delete]

[Front/Back]

[SaveRenderColor] [ClearRenderColor]

[Quadrangulate]

Quadrangulate Angle [   ]

[Select Polygon]   [Deselect Polygon]

[Close]

1) Select polygons to be combined (two or more)

2) Set a threshold angle (alpha)

If (angle formed by the polygon vectors)

    < alpha then combine.

* Polygons cannot be combined unless they have identical materials (color, texture ID, UV).

Material attenuation

A metallic feel can be represented by using a steep light attenuation curve.

* Is not included in the current libgs, libgte

* The load on the GTE is increased

* A "writing of the drawing color" is possible. This does not increase the load on the GTE.

Automatic placement of TIM data

* TIM data is placed automatically in VRAM

* Position info for the TIM file can be saved using "Save TIM position"

Specifying texture regions

1) Use "Specify texture region" to select a portion of the texture data
2) Use "Automatic" to have the size and position of the region adjusted automatically.

Texture mapping

Fine adjustments to texture

^

< >

V
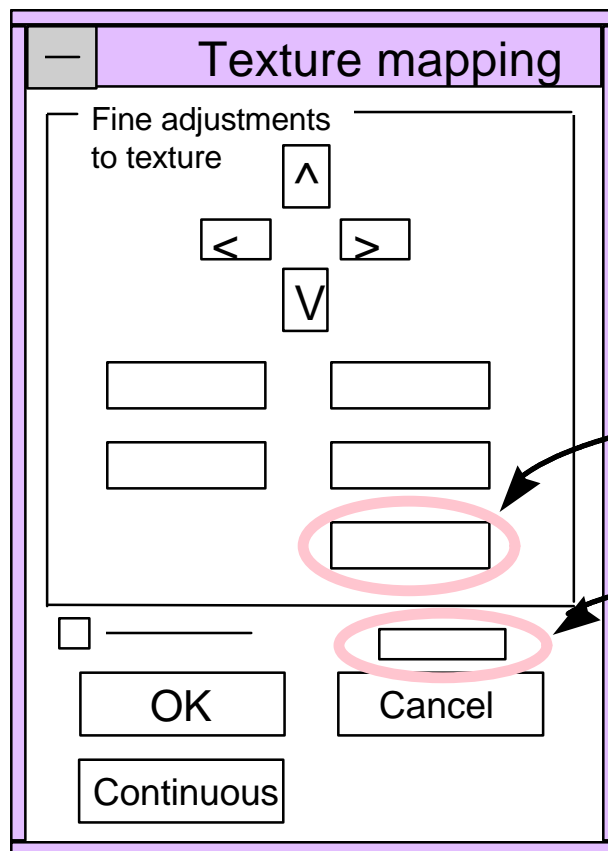
Automatic

Specify region

OK   Cancel

Continuous

Fine UV adjustments

Fine adjustments to the UV values for polygon vertices

1) Select the polygon to be adjusted
2) Select the vertex to be adjusted (Alt + drag)
3) Cancel select mode display
4) Adjust UV value (Ctrl + Alt + arrow keys)

Fine adjustments to texture maps, animation of textures

CONFIDENTI
AL

Maximize the main window of the Material Editor

Animation Tool

Topics covered in this seminar

The sequence is all done but ...

* The models need to be rearranged
* The size of a character needs to be readjusted
* A different character having the same motions needs to be created
* The character needs to be left handed

CONFIDENTIAL

In order to rearrange models ...

Models are entered in HRC files by their
RSD filenames

* What happens if an RSD filename is changed in
    an HRC file?
* What happens if a different file having the
    same RSD filename is placed in the RSD directory?
* What happens if the PLY filename in an RSD file
    is renamed?

Rearranging models

#Alias Block
ObjectName   RSDfileName

Changing an RSD filename in an HRC file

* Use an editor to open the HRC file
* Change the RSD filename in the Alias Block

Problems and Solutions

* What should be done if the size of a part changes?
Export/Import the CAP file yourself

CONFIDENTIAL

In order to adjust the size of a character...

The PlayStation scale is inherited

  * What happens when the top-level scale
    of a character is changed?

Preparation: Make an anchor

  * Make an origin at the top level of the character
    hierarchy
  * Put the origin at the origin of the world coordinate
    system and do not move it
  * Do not make key frames

Adjusting the size of a character

Override Parameters Block
ObjectName JointCode   Layer#   ColorCode
Rx Ry Rz   Sx Sy Sz   Tx Ty Tz

Changing the Override Parameters Block

* Use an editor to open the HRC file
* Change the scale of the anchor
* Open the HRC, sequence
* Select the anchor and create key frame
* Export to TOD

Creating a different character with the same motions

Step 1   Duplicate

Step 2   Copy the character (HRC) and build it up as a new character

Step 3   Import CAP

The Duplicate command

Parameter settings

* Name of new object
* Name of new sequence
    * Saved as a CAP file
    * If all the joints are set to the origin,
      eliminate the parameter by setting the Origin(s)
      Only check box
* HRC filename

Use the Check button to check for name conflicts

Use the Duplicate button to execute

CONFIDENTIAL

Making a character left handed

Preparation: RSDFORM.EXE

Since the model is to be left/right mirrored, RSDFORM.EXE needs to be in the C:\PSXGRAPH\BIN\ directory

Step 1 Duplicate

Step 2 Import CAP

Making a character left handed

Turn the Mirroring switch On in the Duplicate dialog box

To make a left/right mirror of a model,
specify a new RSD filename

Problems and solutions

When a model is mirrored, its textures are reversed as well.
If an RSD filename is not specified (left blank),
the textures are not reversed. Do not reverse objects that
are not on the plane of symmetry (e.g. the head, torso, hip).

Give and take of sequences after Duplicate operation

Characters can give and take sequences
via CAP files

Set Look Up OST check box to ON
(Import CAP dialog box)

If an OST file made during the Duplicate
operation is present, the receiver can
automatically convert the object name.

**CONFIDENTI
AL**

MIMe Animation

Topics

1. Principles of MIMe animation

2. Procedure for creating MIMe animation data
   - Necessary tools and data format
   - Basic operation of the tools
   - Issues to note in creating data

3. MIMe applications

4. Conclusion
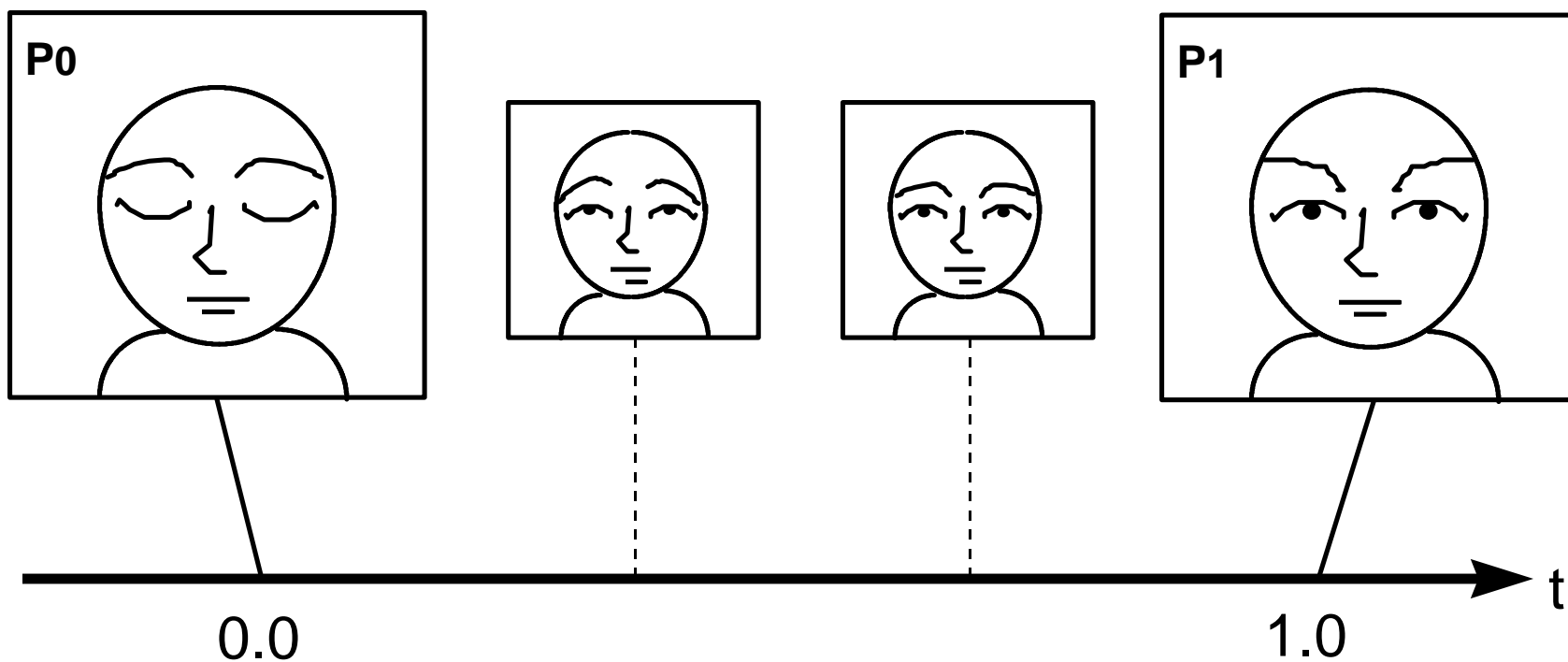
1. Principles of MIMe animation

Interpolation

$$P(t) = (1-t)*P_0 + t*P_1 \quad (0<t<1)$$

P0

P1

0.0

1.0

t

Multiple Inbetweening Method

$$P(t) = \acute{E}_{i=0}^{n}{\scriptstyle\infty}\ W_i(t) * P_i \qquad (\acute{E}{\scriptstyle\infty}_{i=0}^{n}\ W_i(t) = 1.0)$$

P0

P1

P2

P3

-1   0   1

-1   0   1

-1   0   1

0.5

0.5   **-1.0**   **1.0**

+

MIMe Animation

basic model

deformed model

-1   0   1   -1   0   1   -1   0   1

-1   0   1   -1   0   1   -1   0   1

-1   0   1   -1   0   1   -1   0   1

deformed model - basic model = difference vector

changes in composite ratio of difference vectors
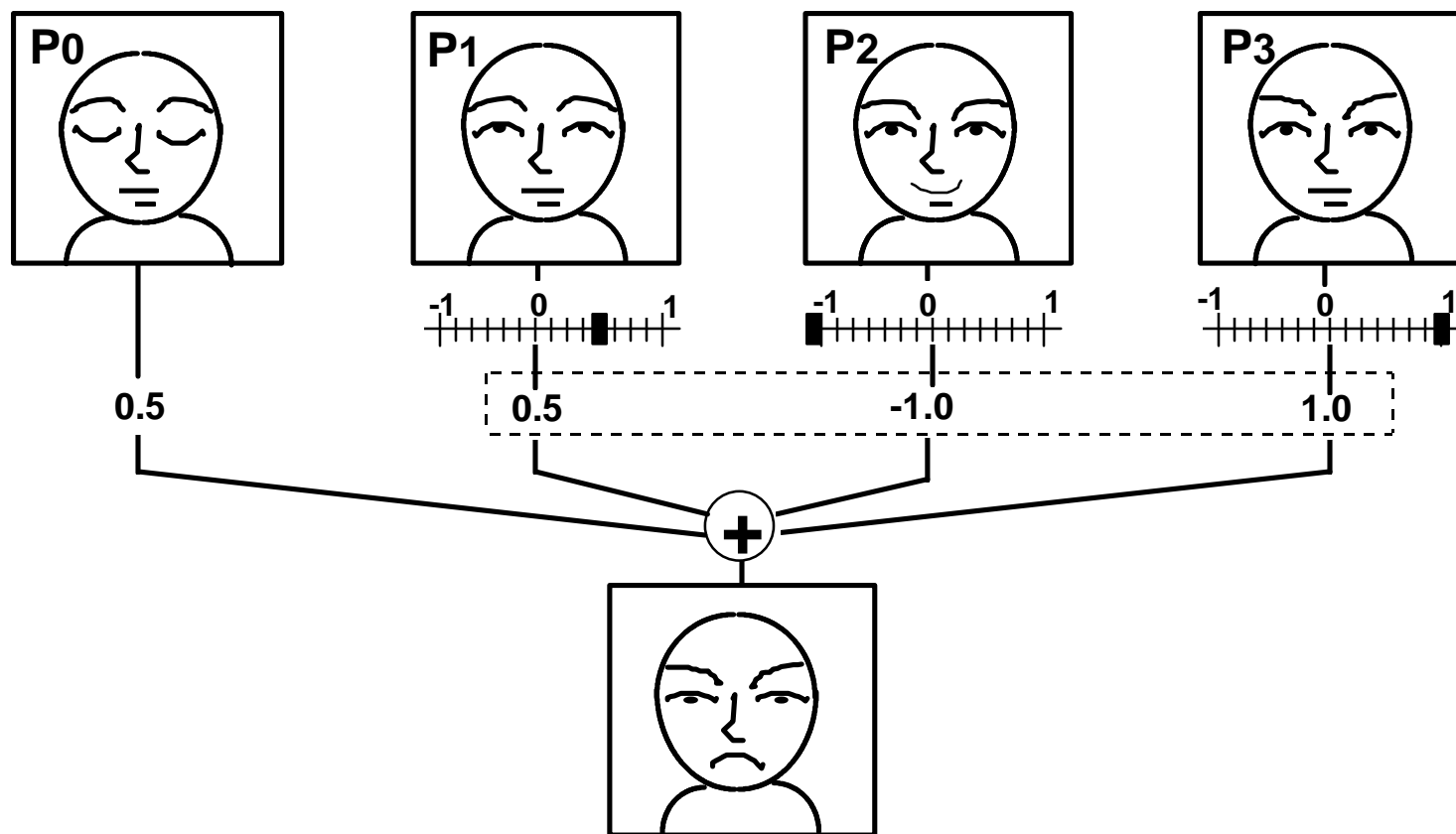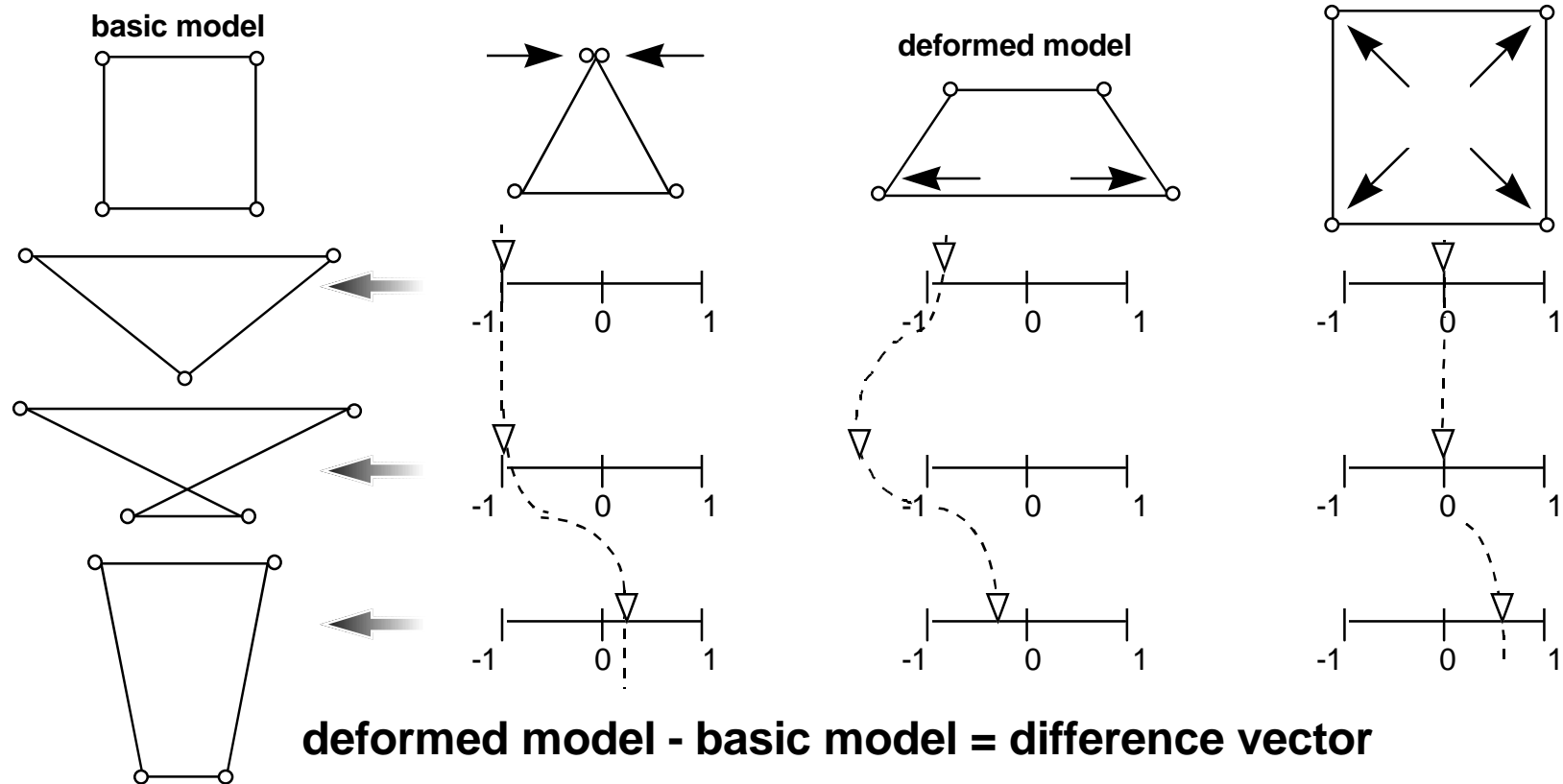
waveform basic model + Sigma (difference x waveform) = MIMe animation

Key frame animation and MIMe animation

- Main point is how the deformed model is selected

- Motion is based on variations in waveforms

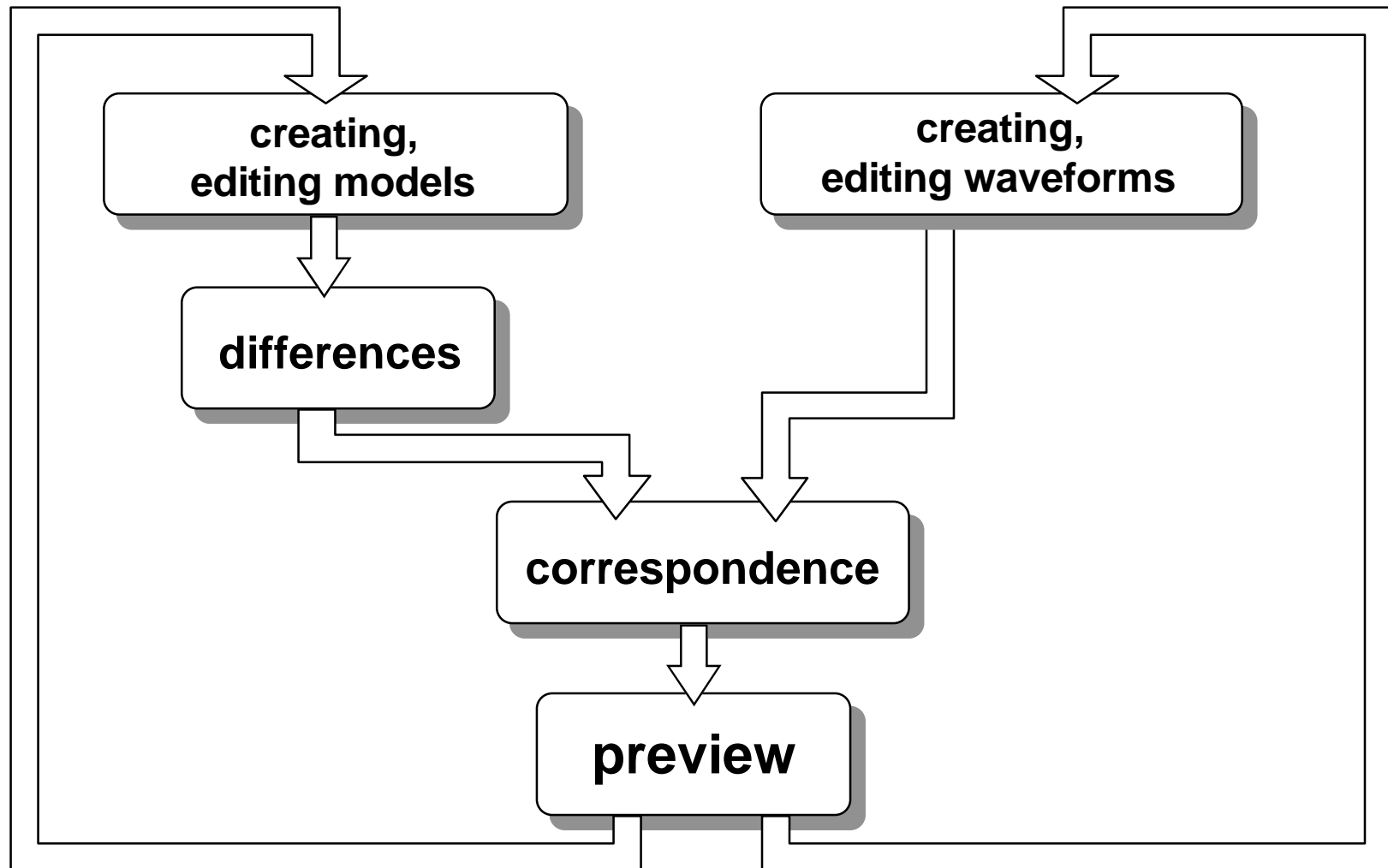# 2. Procedure for creating MIMe animation data

Flowchart for creating MIMe animation

CONFIDENTIAL

(1) Creating, editing models

Data: DXF, RSD, TMD

Tools: 3D modeler, dxf2rsd, dxf2rsdw, rsdlink

Points

- Vertices cannot be increased or decreased

- The ordering of vertices cannot be changed

- Use triangles

(2) Creation of differences
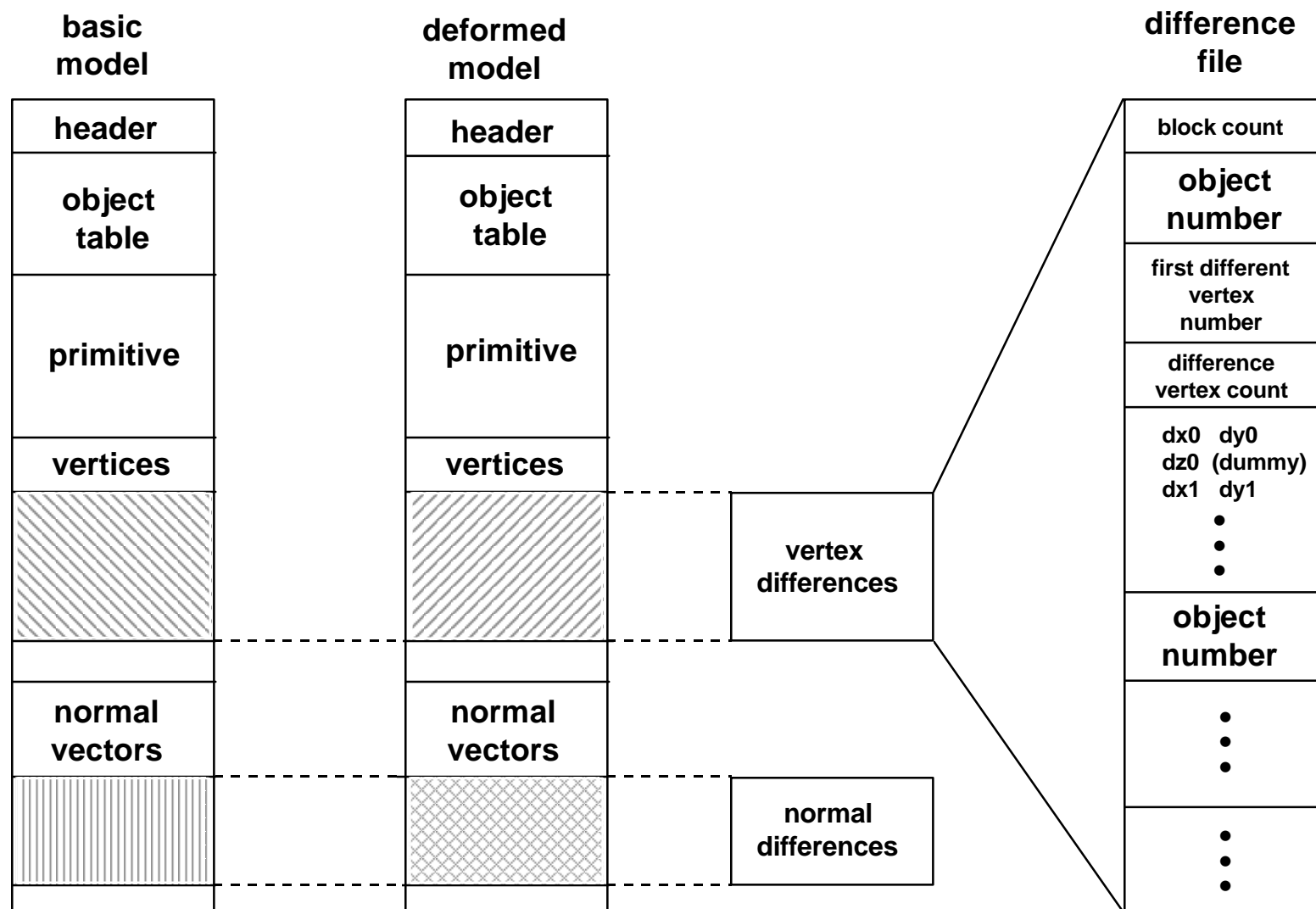
Data: TMD, VDF, NDF

Tools: mimefilt, mimesort

Points

- Only vertices (and normal vectors) are needed
- The texture is needed only for the basic model
- Difference data formats (VDF, NDF)
- Optimization of difference data (mimesort)

Format for difference data

basic model	deformed model	difference file
header	header	block count
object table	object table	**object number**
primitive	primitive	first different vertex number
vertices	vertices	difference vertex count

vertex differences

normal differences

difference file:

block count

object number

first different vertex number

difference vertex count

dx0   dy0
dz0   (dummy)
dx1   dy1
•
•
•

object number

•
•
•

•
•
•

basic model: header, object table, primitive, vertices, normal vectors

deformed model: header, object table, primitive, vertices, normal vectors

(3) Creating waveforms

Data: RSD or TMD , WVF , (MWD)

Tool: mimewave

- mimewave functions
  - **editing waveforms**
  - **assigning correspondences between the deformed model and the waveform**
- explanation of the screen
- creating a waveform
  - Bezier curves
- editing waveforms
  - **continuation of waveforms**
  - **attribution transformations**
  - **insertion and deletion of control points**
- saving waveform data
- waveform data format

CONFIDENTIAL

Waveform data format (WVF)

```
#
MODELNAME= c:\psxgraph\data\boxer01.tmd /* full path name of model file */
MIMEID= 0                               /* position in differential file */
WAVEID= 0                      /* wave ID; common waves have the same ID */ OFFSET= 0
/* starting wave frame */
WAVELENGTH= 80                 /* overall frame count */
DATASIZE= 66                /* wave sample count */
0                                   /* sample data */
0
0
...

MODELNAME= c:\psxgraph\data\boxer02.tmd
MIMEID= 1
WAVEID= 1
OFFSET= 0
WAVELENGTH= 80
DATASIZE= 80
10
270
933
...
```

Confirmation by previewing

Data: TMD, VDF, NDF, TIM, binary waveform data

Tools: mimewave, wavelink, preview (mimewave.cpe)

- preview with PlayStation board
- batch file/make file
- invalid model settings
- changing basic models
- entering commands
- adjusting the preview

    - normal vector MIMe (preview substitution)
    - if texture is present (batch file editing)

Batch file

The batch file below is created using the Save batch file command in MIMeWave

```
resetps

@REM MIMeWave changes lines below here. -- DO NOT EDIT THIS LINE!

set BASE=boxer00
set WAVE=boxer

mimefilt -n boxer00 boxer01 boxer02 boxer03 boxer04
wavelink -o boxer.dat boxer.wvf

@REM MIMeWave changes lines above here. -- DO NOT EDIT THIS LINE!

call patch.bat

pqbload %BASE%.tmd 80100000
pqbload %BASE%.vdf 80180000
pqbload %BASE%.ndf 80200000
pqbload %WAVE%.dat 80400000

@REM You need ONE concatenated texture file with an end mark ('0').
@REM copy /b a.tim+b.tim+c.tim textures.tim
del textures.tim
echo 0 >> textures.tim
pqbload textures.tim 80500000

run c:\psxgraph\bin\mimewave.cpe
```

The other sections are created when the batch file is first created. Afterwards, they stay unchanged so necessary processing can be added. Please do not modify these two comment lines.

Make file

The make file below is created when the mak extension is chosen from the Save batch file command in MIMEWave

```
MIMEFILT = mimefilt -n
WAVELINK = wavelink
RSDLINK = rsdlink
LOADER = pqbload
RESET = resetps
PATCH = patch.dat
RUN = run


PROGRAM = c:\psxgraph\bin\mimewave.cpe
WAVEDATA = boxer.dat
# TEXTURES = a.tim+b.tim+c.tim
TIMFILE = textures.tim
TMDADDR = 80100000
VDFADDR = 80180000
NDFADDR = 80200000
WAVADDR = 80400000
TEXADDR = 80500000

run : load
    $(RUN) $(PROGRAM)

# MIMeWave changes lines below here. --DO NOT EDIT THIS LINE!

MAKEFILE = makefile.mak
BASE = boxer00
WAVE = boxer
WVF  = boxer.wvf
```

```
boxer.dat : $(WVF)
    $(WAVELINK) -o boxer.dat $(WVF)

boxer00.vdf : boxer00.tmd boxer01.tmd ... boxer04.tmd $(MIMEDEPEND)
    $(MIMEFILT) boxer00 boxer01 boxer02 boxer03 boxer04

# MIMeWave changes lines above here. -- DO NOT EDIT THIS LINE!

reset :
    $(RESET)
    $(PATCH)

load : reset $(BASE).tmd $(BASE).vdf $(BASE).ndf $(WAVEDATA)
$(TIMFILE)
    $(LOADER) $(BASE).tmd $(TMDADDR)
    $(LOADER) $(BASE).vdf $(VDFADDR)
    $(LOADER) $(BASE).ndf $(NDFADDR)
    $(LOADER) $(WAVEDATA) $(WAVEADDR)
    $(LOADER) $(TIMFILE)  $(TEXADDR)
    del $(TIMFILE)

textures.tim :
# You need ONE concatenated texture file with an end mark('0').
    echo 0 > $(TIMFILE)
#    copy /b $(TEXTURES) $(TIMFILE)
#    echo 0 >> $(TIMFILE)
```

Textured models

For textured models, it is necessary to modify the batch file or the make file.
In either case, a temporary file (textures.tim) must be created that links all the
necessary texture files. A mark ('0') is added to the end of the linked file so that
the end can be detected when it is loaded.

Change the batch file or the make file as shown below

For batch files (where a.tim, b.tim, c.tim are the textures used)

```
@REM You need ONE concatenated texture file with an end mark ('0').
copy /b a.tim+b.tim+c.tim textures.tim
echo 0>> textures.tim
pqbload textures.tim 80500000
```

For make files (where a.tim, b.tim, c.tim are the textures used)

```
TEXTURES = a.tim+b.tim+c.tim
TIMFILE  = textures.tim
...
textures.tim :
# You need ONE concatenated texture file with an end mark ('0').
    copy /b $(TEXTURES) $(TIMFILE)
    echo 0>> $(TIMFILE)
```

3. MIMe applications

MIMe with compression

Creating control waveforms

Creating real-time waveforms with the pad

Sample program (TMDVIEW$MIME$)

Rotation movements

Increase deformed models

Interpolate with angles

Spline interpolation

Interpolation of matrices

CONFIDENTIAL

Tuning up the data

Optimization using mimesort

Compression of waveform data

Model tricks

Various MIMes

Angle MIMe

Two-dimensional MIMe

Texture MIMe (UV MIMe)

Color MIMe
...
...

4. Conclusion

Conclusion

MIMe is a real time image synthesis technology

... by optimizing models and varying waveforms

CONFIDENTIAL

3D Studio TOD Plug-in

Topics

The TOD plug-in program is a 3D Studio R4 plug-in for the PlayStation.

* Save object data (RSD)

* Save animation data (TOD)

* Preview TOD data using DTL-H2000

Save RSD data

* Save using local coordinates for each object
* Automatic generation of object names
* Three-sided polygons only (four-sided polygons cannot be generated)
* Conversion of 3DS smoothing info to RSD
* Diffuse reflection lighting from 3DS is converted to RSD polygon colors
* Conversion of UV values for texture maps
* No conversion of texture image data

Save TOD data

* Save animation of objects (translations, rotations, scaling) and camera animation (translations, rotations) to TOD

* Morphing cannot be converted

* Light source info cannot be converted

scale, Frames from/to, TOD type

scale

scale (3DS coordinates) * (scale value) = (RSD, TMD coordinates)

Frames from/to

Specify the range of frames to be saved

TOD type

Matrix: Matrix type

RST: RST type

* Matrix type requires less processing but is not suited for interframe interpolation on the PS.

Hierarchy

on: Saves coordinate system of hierarchy
structure to TOD

off: Not saved

* Not saving the hierarchy structure reduces the amount of calculations done on the PS, but this is not suited for cases where an object is being moved interactively.

Pivot

Pivot

on: Pivot info is reflected in the TOD

off: Pivot info is not reflected

On (rotation around pivot is local) = (rotation around origin)

Off (rotation around pivot is local) = (rotation around origin) + (translation)

PlayStation TOD Plug-in

Scale:	
Frames from:	
to:	
TOD type:	Matrix RST
Hierarchy:	on off
Pivot:	on off
Frames:	All 1/2 1/3 1/4 1/5

Preview	Quick Preview
Save RSD	Save TOD

Frames

| Frames | Frames can be skipped and saved. |

All: All frames are saved
1/2: Every other frame is saved (0,2,4...)
1/3: Every third frame is saved (0,3,6...)
1/4: Every fourth frame is saved (0,4,8...)
1/5: Every fifth frame is saved (0,5,10...)

*** Skipping frames can reduce the size of data but
will require the PS to interpolate between frames.**