Media Field Navigation

Selective Shared Immersive Reality Substitution

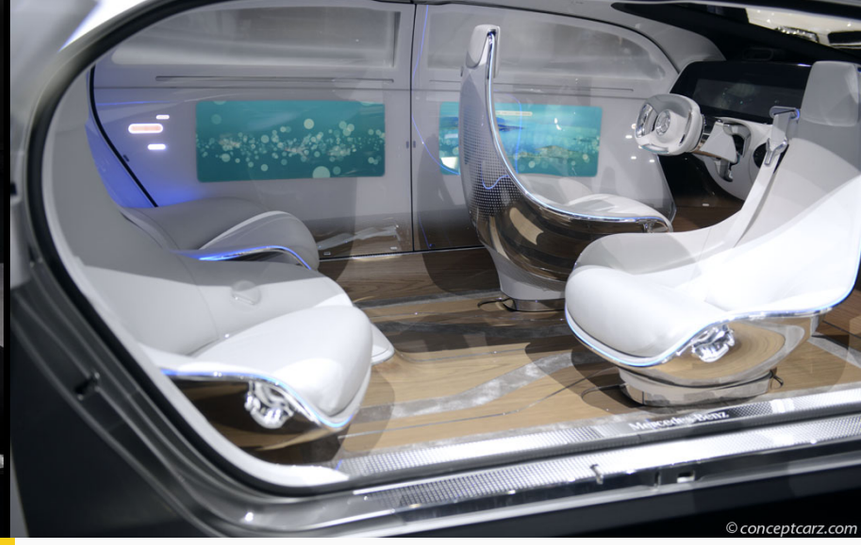
FORUM8 | World16

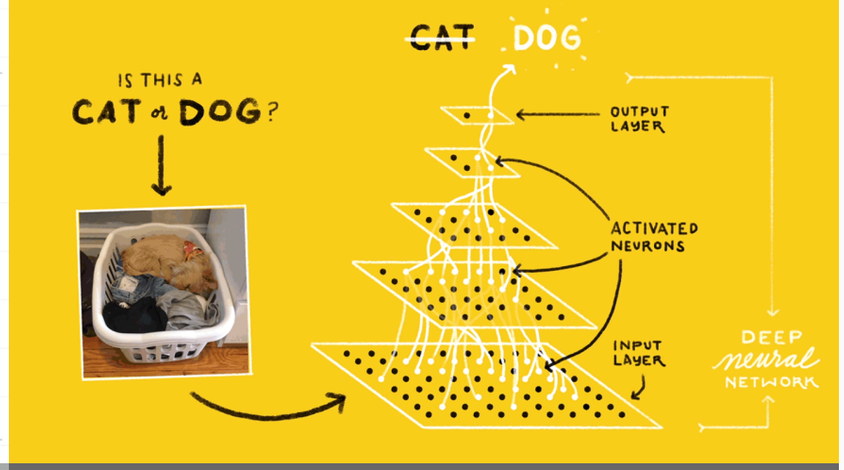
Marcos **Novak**
transLAB
MAT/UCSB

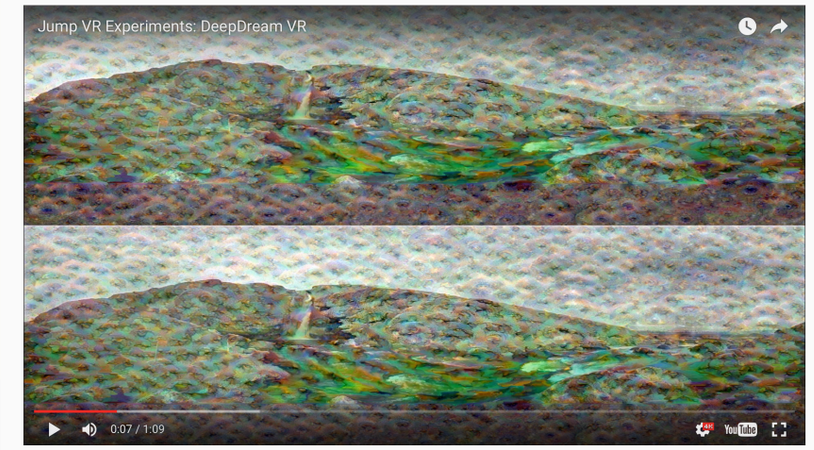
Tokyo
November 2016

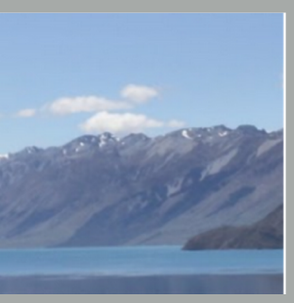
Overview

- MFN (Media Field Navigation): MFN refers to transportation as navigation through fields of information and media.
- SSIRS (Selective Shared Immersive Reality Substitution): SSIRS refers to the substitution of a selectively modified immersive reality in place of another, with the purpose of altering the original MFN.

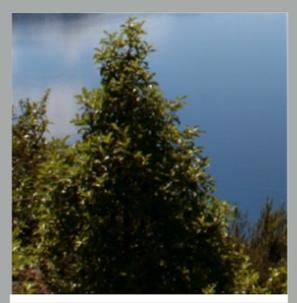






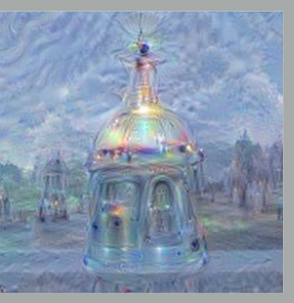


Horizon

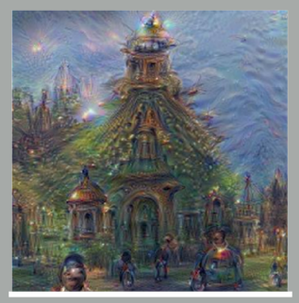


Trees

Leaves

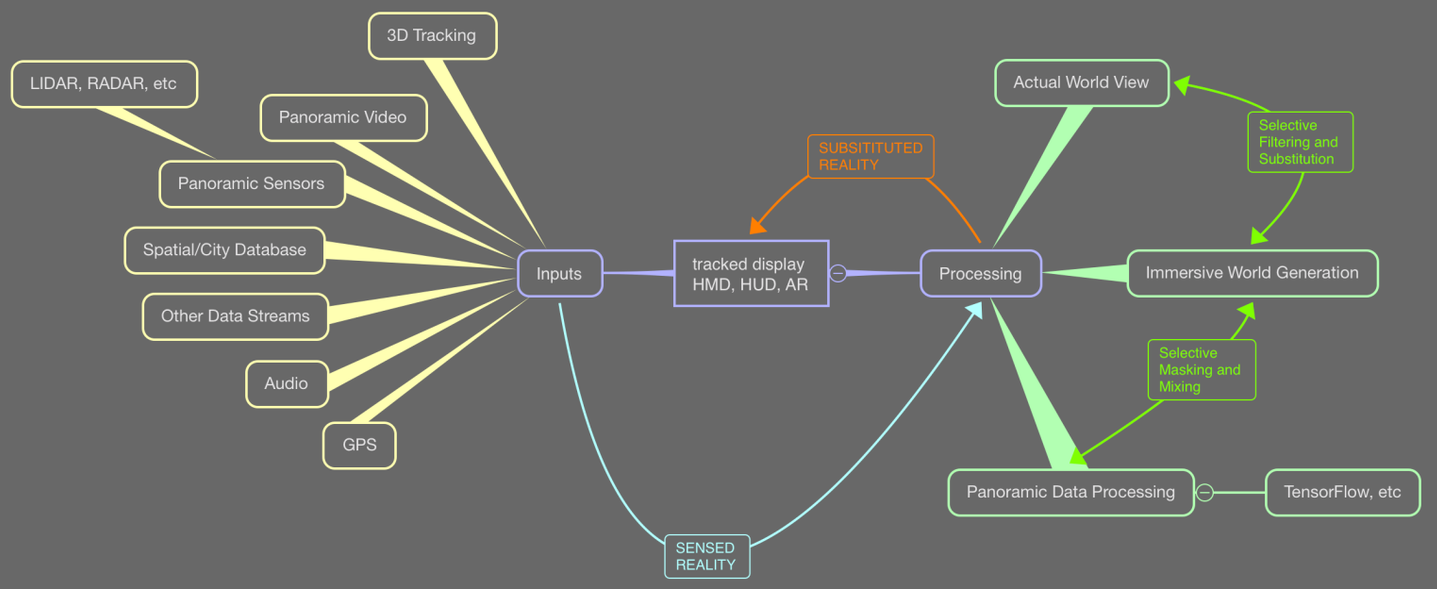


Towers & Pagodas



Buildings

Birds & Insects



Media Field Navigation

Selective Shared Immersive Reality Substitution

Pipeline: sources, processes, filters, shaders, outputs

- Sources -> sensors, video, simulation, databases
- Processes -> object and facial recognition, machine vision
- Filters -> 2D and 3D filters, neural nets, video, point clouds
- Shaders -> GPU shaders for real-time processing
- Outputs -> Screens, HUDs, HMDs, AR, MR, VR

Software: Mathematica, Deep Dreamer, Max, GPU Shaders

- Mathematica
- Deep Dreamer
- Max
- GPU Shaders

Mathematica

- Wolfram Language
- Computer Vision
- Image Identify
- FindFaces

Deep Dreamer

- Deep Dream Generator
- Computer Vision
- Image Identify
- FindFaces

Max

Real time video audio and video processing of input streams using matrix operators.

- jit.op

jit.op.maxhelp (unlocked)

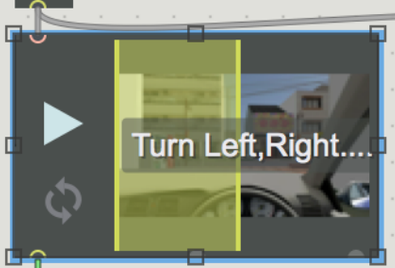
100%

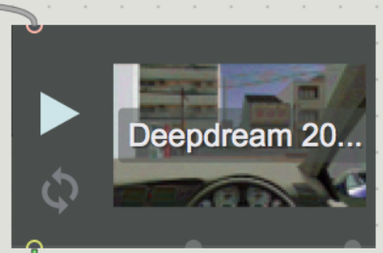
basic | operators | using multiple operators | ?

jit.op

Apply binary or unary operators

The jit.op object applies either a binary operator to two input matrices, or a unary operator to the left input matrix. A different operator may be specified for each plane, or a scalar may be specified via the val attribute as an alternate to using a second matrix.





==

op | min | getop

0.

You can use a scalar, too

jit.op @op * @val 0.5

Arguments:

supports: any type/planecount.

print @popup 1

9.86103

fps

jit.pla...

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

jit.op

arithmetic

pass: pass left input, no operator;
*: multiplication;
/: division;
+: addition;
-: subtraction;
+m: addition modulo (char only);
-m: subtraction modulo (char only);
%: modulo;
min: minimum;
max: maximum;
abs: absolute value (unary);
avg: average;
absdiff: absolute value of difference;
!pass: pass right input, no operator;
!/: right input/left input (flipped);
!-: right input-left input (flipped);
!%:right input/left input (flipped);
ignore: leave previous output value;
fold: mirrored modulo (float32/float64 only);
wrap: positive modulo (float32/float64 only);

trigonometric

(float32/float64 only, unary unless otherwise mentioned)

sin: sine;
cos: cosine;
tan: tangent;
asin: arcsine;
acos: arccosine;
atan: arctangent;
atan2: arctangent (binary);
sinh: hyperbolic sine;
cosh: hyperbolic cosine;
tanh: hyperbolic tangent;
asinh: hyperbolic arcsine;
acosh: hyperbolic arccosine;
atanh: hyperbolic arctangent;

bitwise

(long/char only)
&: bitwise and;
|: bitwise or;
^: bitwise xor;
~: bitwise compliment (unary);
>>: right shift;
<<: left shift

logical

&&: logical and;
||: logical or;
!: logical not (unary);
>: greater than;
<: less than;
>=: greater than or equal to;
<=: less than or equal to;
==: equal;
!=: not equal

>p: greater than (pass);
<p: less than(pass);
>=p: greater than or equal to (pass);
<=p: less than or equal to (pass);
==p: equal (pass);
!=p not equal (pass)

exp/log/rounding

(float32/float64 only, unary unless otherwise mentioned)
exp: e to the x;
exp2: 2 to the x;
ln: log base e;
log2: log base 2;
log10: log base 10;
hypot: hypotenuse (binary);
pow: x to the y (binary);
sqrt: square root;
ceil: integer ceiling;
floor: integer floor;
round: round to nearest integer;
trunc: truncate to integer

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

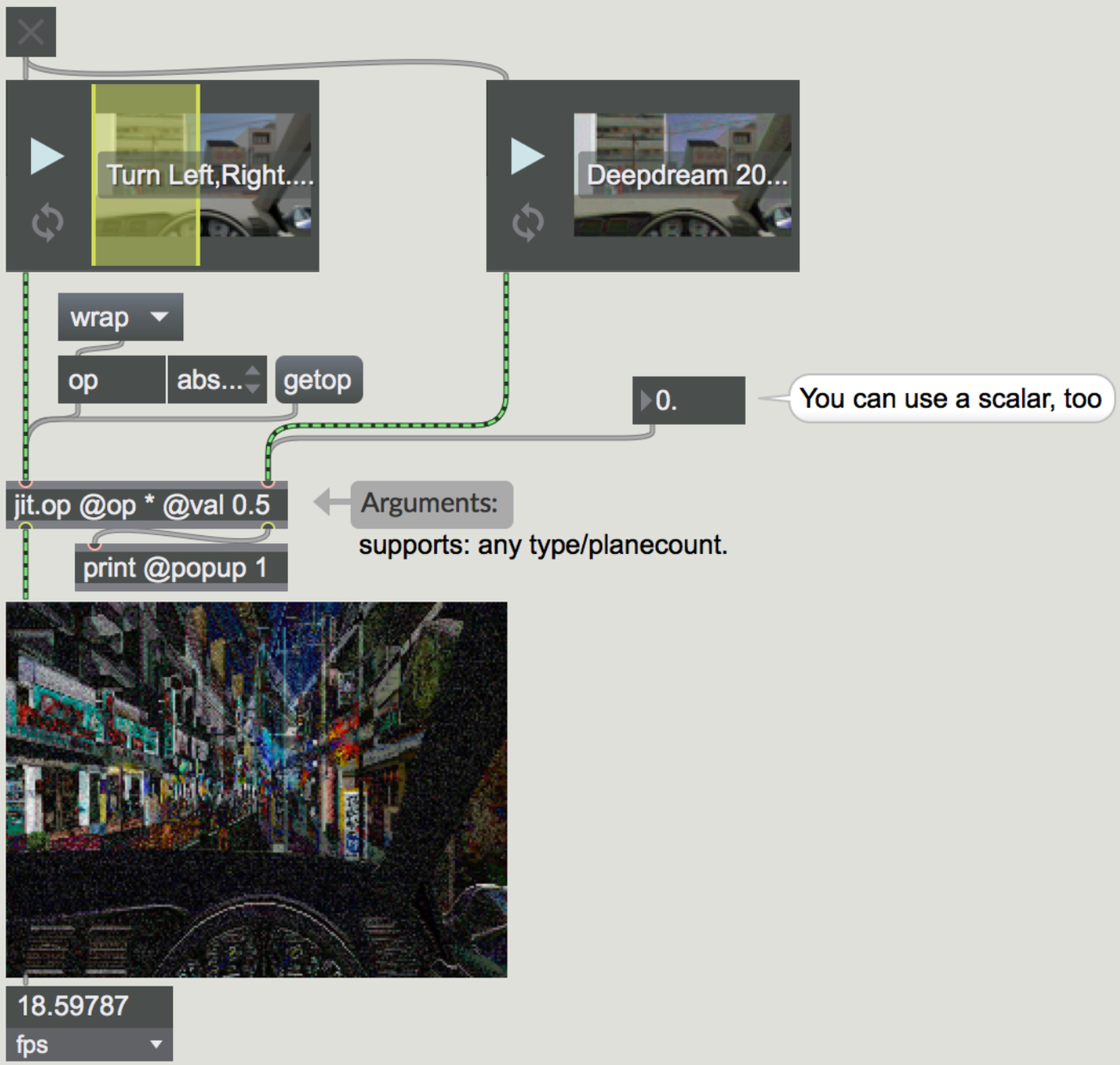
- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

jit.op

Apply binary or unary operators

The jit.op object applies either a binary operator to two input matrices, or a unary operator to the left input matrix. A different operator may be specified for each plane, or a scalar may be specified via the val attribute as an alternate to using a second matrix.



jit.pla...

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

GPU Shaders

Real-time manipulation via custom shaders applied during scene rendering and post-processing.

- OpenGL
- GLSL
- HLSL
- Metal
- others

MeshLab_64bit v1.3.3

Project_1

FOV: 60
FPS: 270.3

Mesh: Laurana50k.ply
Vertices: 27861
Faces: 49954
VC

MeshLab File Edit Filters Render View Windows Help

100% Thu 9:06 AM marcos

Project_1

Form

Uniform Variables | Textures | OpenGL Status | Vertex Program | Fragment Program

```
//Create uniform variables for lighting to allow user interaction
uniform float SpecularContribution;
uniform vec3 LightPosition;

varying vec3 MCPosition;
varying float LightIntensity;

void main(void)
{
   float diffusecontribution  = 1.0 - SpecularContribution;

   // compute the vertex position in eye coordinates
   vec3  ecPosition          = vec3(gl_ModelViewMatrix * gl_Vertex);

   // compute the transformed normal
   vec3  tnorm               = normalize(gl_NormalMatrix * gl_Normal);

   // compute a vector from the model to the light position
   vec3  lightVec            = normalize(LightPosition - ecPosition);

   // compute the reflection vector
   vec3  reflectVec          = reflect(-lightVec, tnorm);

   // compute a unit vector in direction of viewing position
   vec3  viewVec             = normalize(-ecPosition);

   // calculate amount of diffuse light based on normal and light angle
   float diffuse             = max(dot(lightVec, tnorm), 0.0);
   float spec                = 0.0;

   // if there is diffuse lighting, calculate specular
   if(diffuse > 0.0)
```

Close Shader

FOV: 60
FPS: 72.5

Mesh: Laurana50k.ply
Vertices: 27861
Faces: 49954
VC

QUESTIONS

- what is “signal” and what is “noise”?
- how can we increase “signal” and reduce “noise”?
- what elements of the environment cannot be ignored?
- what elements of the environment can be modified?
- how can transitions between actual and modified be handled?

jit.op.maxhelp (unlocked)

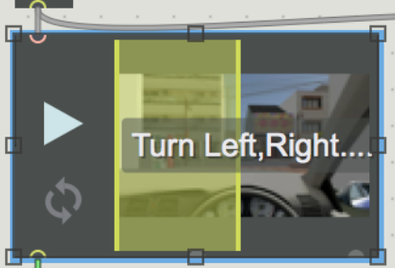
100%

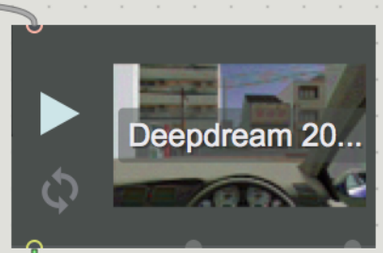
basic | operators | using multiple operators | ?

jit.op

Apply binary or unary operators

The jit.op object applies either a binary operator to two input matrices, or a unary operator to the left input matrix. A different operator may be specified for each plane, or a scalar may be specified via the val attribute as an alternate to using a second matrix.





==

op | min | getop

0.

You can use a scalar, too

jit.op @op * @val 0.5

Arguments:

supports: any type/planecount.

print @popup 1

9.86103

fps

jit.pla...

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

jit.op

arithmetic

pass: pass left input, no operator;
*: multiplication;
/: division;
+: addition;
-: subtraction;
+m: addition modulo (char only);
-m: subtraction modulo (char only);
%: modulo;
min: minimum;
max: maximum;
abs: absolute value (unary);
avg: average;
absdiff: absolute value of difference;
!pass: pass right input, no operator;
!/: right input/left input (flipped);
!-: right input-left input (flipped);
!%:right input/left input (flipped);
ignore: leave previous output value;
fold: mirrored modulo (float32/float64 only);
wrap: positive modulo (float32/float64 only);

trigonometric

(float32/float64 only, unary unless otherwise mentioned)

sin: sine;
cos: cosine;
tan: tangent;
asin: arcsine;
acos: arccosine;
atan: arctangent;
atan2: arctangent (binary);
sinh: hyperbolic sine;
cosh: hyperbolic cosine;
tanh: hyperbolic tangent;
asinh: hyperbolic arcsine;
acosh: hyperbolic arccosine;
atanh: hyperbolic arctangent;

bitwise

(long/char only)
&: bitwise and;
|: bitwise or;
^: bitwise xor;
~: bitwise compliment (unary);
>>: right shift;
<<: left shift

logical

&&: logical and;
||: logical or;
!: logical not (unary);
>: greater than;
<: less than;
>=: greater than or equal to;
<=: less than or equal to;
==: equal;
!=: not equal

>p: greater than (pass);
<p: less than(pass);
>=p: greater than or equal to (pass);
<=p: less than or equal to (pass);
==p: equal (pass);
!=p not equal (pass)

exp/log/rounding

(float32/float64 only, unary unless otherwise mentioned)
exp: e to the x;
exp2: 2 to the x;
ln: log base e;
log2: log base 2;
log10: log base 10;
hypot: hypotenuse (binary);
pow: x to the y (binary);
sqrt: square root;
ceil: integer ceiling;
floor: integer floor;
round: round to nearest integer;
trunc: truncate to integer

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

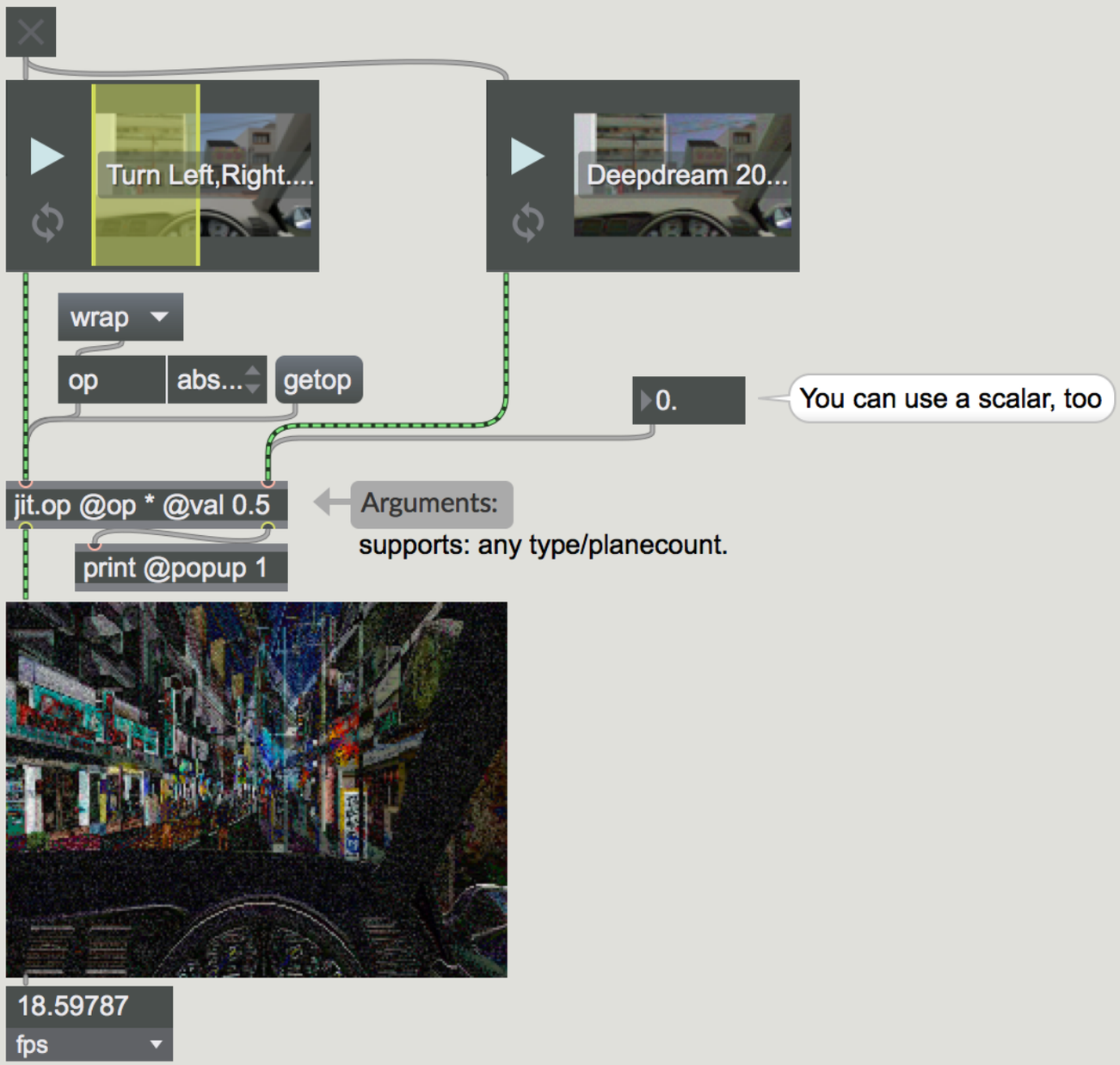
- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

jit.op

Apply binary or unary operators

The jit.op object applies either a binary operator to two input matrices, or a unary operator to the left input matrix. A different operator may be specified for each plane, or a scalar may be specified via the val attribute as an alternate to using a second matrix.



jit.pla...

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

QUESTIONS

- what is the range of possible expressions?
- how can expressions be triggered interactively?
- how would the environment react to several passengers?
- how would other media and information be overlayed?
- how could this approach operate on 3D point clouds in VR?

jit.op.maxhelp (unlocked)

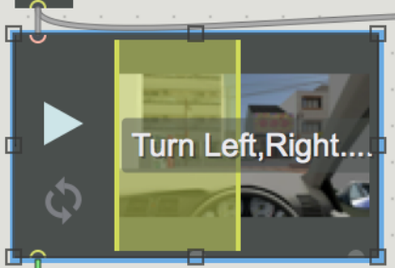
100%

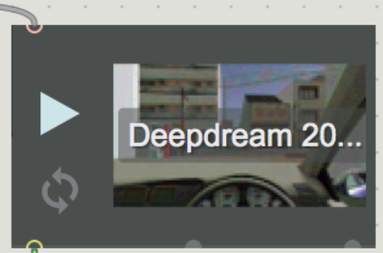
basic | operators | using multiple operators | ?

jit.op

Apply binary or unary operators

The jit.op object applies either a binary operator to two input matrices, or a unary operator to the left input matrix. A different operator may be specified for each plane, or a scalar may be specified via the val attribute as an alternate to using a second matrix.





==

op | min | getop

0.

You can use a scalar, too

jit.op @op * @val 0.5

Arguments:

supports: any type/planecount.

print @popup 1

9.86103

fps

jit.pla...

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

jit.op

arithmetic

pass: pass left input, no operator;
*: multiplication;
/: division;
+: addition;
-: subtraction;
+m: addition modulo (char only);
-m: subtraction modulo (char only);
%: modulo;
min: minimum;
max: maximum;
abs: absolute value (unary);
avg: average;
absdiff: absolute value of difference;
!pass: pass right input, no operator;
!/: right input/left input (flipped);
!-: right input-left input (flipped);
!%:right input/left input (flipped);
ignore: leave previous output value;
fold: mirrored modulo (float32/float64 only);
wrap: positive modulo (float32/float64 only);

trigonometric

(float32/float64 only, unary unless otherwise mentioned)

sin: sine;
cos: cosine;
tan: tangent;
asin: arcsine;
acos: arccosine;
atan: arctangent;
atan2: arctangent (binary);
sinh: hyperbolic sine;
cosh: hyperbolic cosine;
tanh: hyperbolic tangent;
asinh: hyperbolic arcsine;
acosh: hyperbolic arccosine;
atanh: hyperbolic arctangent;

bitwise

(long/char only)
&: bitwise and;
|: bitwise or;
^: bitwise xor;
~: bitwise compliment (unary);
>>: right shift;
<<: left shift

logical

&&: logical and;
||: logical or;
!: logical not (unary);
>: greater than;
<: less than;
>=: greater than or equal to;
<=: less than or equal to;
==: equal;
!=: not equal

>p: greater than (pass);
<p: less than(pass);
>=p: greater than or equal to (pass);
<=p: less than or equal to (pass);
==p: equal (pass);
!=p not equal (pass)

exp/log/rounding

(float32/float64 only, unary unless otherwise mentioned)
exp: e to the x;
exp2: 2 to the x;
ln: log base e;
log2: log base 2;
log10: log base 10;
hypot: hypotenuse (binary);
pow: x to the y (binary);
sqrt: square root;
ceil: integer ceiling;
floor: integer floor;
round: round to nearest integer;
trunc: truncate to integer

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

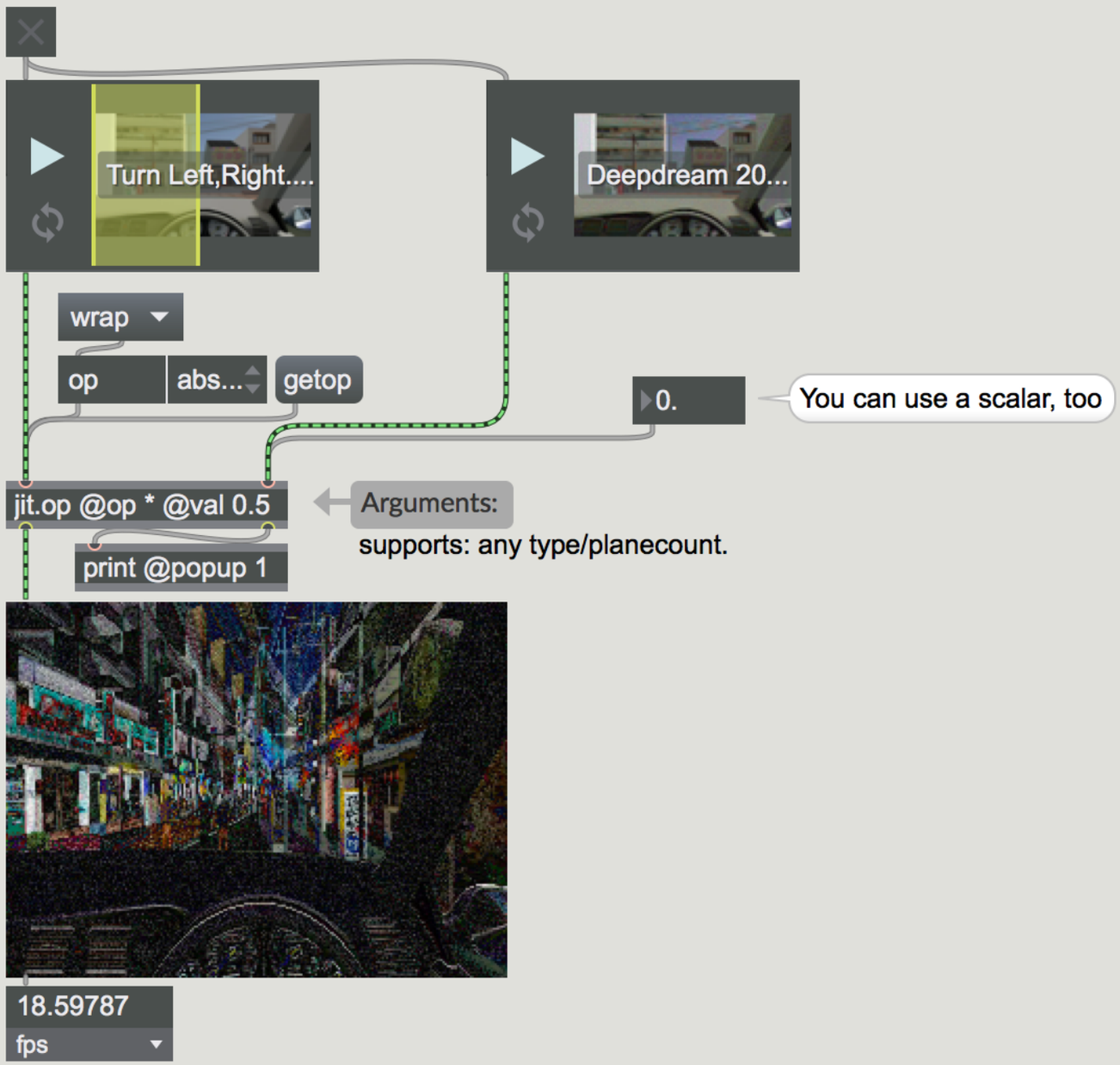
- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

jit.op

Apply binary or unary operators

The jit.op object applies either a binary operator to two input matrices, or a unary operator to the left input matrix. A different operator may be specified for each plane, or a scalar may be specified via the val attribute as an alternate to using a second matrix.



jit.pla...

Play video files

▼ Messages

- int - Start/stop playing a soundfile
- pause - Pause playback
- resume - Resume playback
- clear - Remove all sound files
- rate - Set playback speed for a clip
- loop - Turn looping on/off for a clip
- selection - Select playback endpoi...

▼ Attributes

- displaymode - Display format for ...
- allowreorder - Allow the re-orderi...
- clipheight - Height alloted for eac...
- expansion - Style of accomodatio...
- autopwindow - Automatically pat...

▼ See Also

- playlist~ - Play sound files
- sfplay~ - Play audio file from disk
- waveform~ - buffer~ viewer and ...

Use jit.playlist to organize sets of video files and play them back. Each video is given a visual representation in a clip where a selection from the entire video file may be choosen. Clips may be dragged within a jit.playlist to re-order them, or they maybe

Media Field Navigation

Selective Shared Immersive Reality Substitution

FORUM8 | World16

Marcos **Novak**
transLAB
MAT/UCSB

Osaka
November 2016