NAME

xil_interpolation_table_create, xil_interpolation_table_destroy – create or destroy an interpolation table object

SYNOPSIS

#include <xil/xil.h>

XilInterpolationTable xil_interpolation_table_create (XilSystemState *state*,
unsigned int *kernel_size*,
unsigned int *subsamples*,
float ∗*data*);

void xil_interpolation_table_destroy (XilInterpolationTable *table***);**

DESCRIPTION

These routines create and destroy interpolation table objects. An XilInterpolationTable object is an array of 1xn kernels which represents the interpolation filter in either the horizontal or vertical direction. The datatype of the table is XIL_FLOAT.

The parameter *state* is the XIL system state, *kernel_size* is the size of the kernel, *subsamples* is the number of subsamples between pixels, and *data* is the data of the interpolation table. There is no limit or restrictions on the kernel size or the number of subsamples.

Each subsample requires a separate set of kernel data. Thus, *n* subsamples require *n* ∗ *kernel_size* data elements. For example, a horizontal interpolation table with a kernel size of 7 elements and a pixel subsampling of 3 requires 21 data elements; the first subsample uses the first 7 data elements, the second subsample uses the next 7 data elements, and the third subsample uses the last 7 data elements. If both the horizontal and vertical interpolation tables are NULL, nearest neighbor interpolation is performed.

xil_interpolation_table_destroy () destroys the specified interpolation table object.

ERRORS

For a complete list of XIL error messages by number, consult Appendix B of the *XIL Programmer's Guide.*

EXAMPLES

Create a horizontal interpolation table with seven kernel elements and two subsamples between pixels:

```
XilSystemState state;
XilInterpolationTable horiz_table;
float *data;

horiz_table = xil_interpolation_table_create (state, 7, 2, data);
```

NOTES

The key element in a kernel is the center element; for even-sized kernels, the key elements is the first of the two center elements. Thus, for an 8-element kernel, the key value is the fourth element, which has the array index 3. The key element's array index can be computed as an integer calculation:

```
int array_index = (kernel_size - 1) / 2
```

To preserve the source image's intensity in the destination, an individual kernel's values should sum to one.

SEE ALSO

xil_interpolation_table_get_subsamples(3), **xil_interpolation_table_get_kernel_size**(3), **xil_interpolation_table_get_data**(3), **xil_state_get_interpolation_tables**(3).