NAME

xil_rescale – rescale the data in an image

SYNOPSIS

#include <xil/xil.h>

void xil_rescale (XilImage *src***,**
XilImage *dst***,**
float *∗scale***,**
float *∗offset***);**

DESCRIPTION

This routine performs a pixel-by-pixel rescaling of the data in a *src* (source) image by first multiplying each pixel value by a scale factor and then adding an offset. The result is stored in the *dst* (destination) image. For an n-band image, each array of constants must contain n floats The values in band 0 are scaled by *scale[0]* and added to *offset[0],* and so on.

Pixel values are clipped according to image data type. In this function, a floating point intermediate value is calculated, so clipping/rounding is done after both the multiplication and the addition have occurred.

ERRORS

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

EXAMPLES

Rescale a 4-band short *src* image into the range of XIL_BYTE, and store the result in *dst:*

```
XilImage src, dst;
float scale_values[4], offset_values[4];

src=xil_create(State, 512, 512, 3, XIL_SHORT);
dst=xil_create(State, 512, 512, 3, XIL_SHORT);

/* scale factors for each band */
scale_values[0] = 127.5/32767.0;
scale_values[1] = 127.5/32767.0;
scale_values[2] = 127.5/32767.0;
scale_values[3] = 127.5/32767.0;

/* offset factors for each band */
offset_values[0] = 127.5;
offset_values[1] = 127.5;
offset_values[2] = 127.5;
offset_values[3] = 127.5;

xil_rescale(src, dst, scale_values, offset_values);
```

NOTES

Source and destination images must be the same data type and have the same number of bands. In-place operations are supported.