Unroll Convolution for Interlaced Images

Revision: 1 **INTERNAL BOUNTY**

Creation Date: December 24, 2016 **Awarded To:**

Abstract:
Low level routines for interlaced images have not been as fully optimized as single band images and by association planar images. A benchmark shows interlaced images to be about 4.5x slower than planar images performing the same operation. This bounty is intended to help reduce that gap in performance by unrolling certain convolution operations.

Completion Criteria:

- See Bounties page for general rules/conditions
- Creation of autogenerated unrolled implementations of the following classes
  - ConvolveImageStandard_IL
- Unrolled code shall support
  - Images with 2, 3, and 4 bands/channels
  - Kernels with a width of 3, 5, and 7
  - All the same image types as the original code
- Unrolled code will produce output that is identical to the original code to within floating point tolerance
  - Existing unit tests should be used for this test
- Similar code style and naming that was used for single band images should be used here
  - We can help you with refactoring at this point easily enough
  - We will also help you integrate the unrolled code in with rest of the code base
- Performance improvement by 2x to 5x is expected

Comments:

The most challenging aspect of this task is understand how autocode generation works in BoofCV. Fortunately there is plenty of examples for how to do it. For this task the following is where you should start:

- Unrolled single band images
- Convolution with interleaved images

It's recommended that you first unroll one operation then unit test it and test its runtime performance to make sure everything is going as expected. Unit tests also need to take advantage of the existing infrastructure. This will automate most of the tests through use of

reflections. You will really just need to change the name of a few variables. Again we can help you here.

- Example of a unit test for an unrolled single band image