
ComposeRGBAImageFilter

Release 1.0

Dan Mueller¹

May 11, 2007

¹Queensland University of Technology, Brisbane, Australia

Abstract

This article presents the `itk::ComposeRGBAImageFilter`.

Keywords: ITK, Compose

1 Introduction

The existing `itk::ComposeRGBImageFilter` allows users to compose a multi-channel (RGB) image from three scalar inputs. Currently its RGBA counterpart does not exist. This article presents the `itk::ComposeRGBAImageFilter`.

2 Implementation

A number of additions to the ITK Common namespace must be made to accommodate the proposed `itk::ComposeRGBAImageFilter`:

1. Add an `#include "itkNumericTraitsRGBAPixel.h"` to the bottom of `itkRGBAPixel.h`.
2. Add `itkNumericTraitsRGBAPixel.h` and `itkNumericTraitsRGBAPixel.cxx` to the Common directory.
3. Add `itkNumericTraitsRGBAPixel.cxx` to the list of sources files in `Common/CMakeLists.txt`.

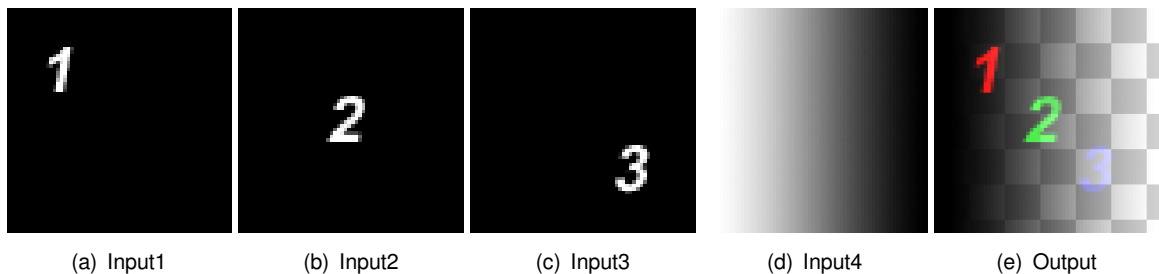


Figure 1: Results of the `ComposeRGBA` test.

3 Code

```

1  /*=====
2
3  Program:   Insight Segmentation & Registration Toolkit
4  Module:    $RCSfile: itkComposeRGBAImageFilter.h,v $
5  Language:  C++
6  Date:      $Date: 2007/05/10 07:00:00 $
7  Version:   $Revision: 1.0 $
8
9  Copyright (c) Insight Software Consortium. All rights reserved.
10 See ITKCopyright.txt or http://www.itk.org/HTML/Copyright.htm for details.
11
12     This software is distributed WITHOUT ANY WARRANTY; without even
13     the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
14     PURPOSE. See the above copyright notices for more information.
15
16  =====*/
17 #ifndef __itkComposeRGBAImageFilter_h
18 #define __itkComposeRGBAImageFilter_h
19
20 #include "itkNaryFunctorImageFilter.h"
21 #include "itkRGBAPixel.h"
22 #include "itkNumericTraitsRGBAPixel.h"
23
24 /** \class ComposeRGBAImageFilter
25  * \brief Implements pixel-wise composition of an RGBA pixel from four scalar images.
26  *
27  * This filter receives four scalar images as input. Each image containing
28  * one of the RGBA components of a color image. The filter produces as output an
29  * RGBA image in which the four components have been unified. The Component
30  * type is preserved from the PixelType of the input images.
31  *
32  * \ingroup IntensityImageFilters
33  */
34
35 namespace itk
36 {
37
38 namespace Functor {
39
40 template< class TInput >
41 class ComposeRGBA
42 {
43 public:
44     typedef RGBAPixel<TInput> OutputType;
45     ComposeRGBA() {}
46     ~ComposeRGBA() {}
47     bool operator!=( const ComposeRGBA & ) const
48     {
49         return false;
50     }
51     bool operator==( const ComposeRGBA & other ) const
52     {
53         return !(*this != other);
54     }
55     inline OutputType operator()( const std::vector< TInput > & in )
56     {

```

```

57     OutputType pixel;
58     pixel.Set( in[0], in[1], in[2], in[3] );
59     return pixel;
60 }
61 };
62 }
63
64 template <typename TInputImage,
65           typename TOutputImage=
66           Image< RGBAPixel< ITK_TYPENAME TInputImage::PixelType >,
67               ::itk::GetImageDimension<TInputImage>::ImageDimension > >
68 class ITK_EXPORT ComposeRGBAImageFilter :
69     public
70     NaryFunctorImageFilter<TInputImage,TOutputImage,
71                           Functor::ComposeRGBA< ITK_TYPENAME TInputImage::PixelType > >
72 {
73 public:
74     /** Standard class typedefs. */
75     typedef ComposeRGBAImageFilter Self;
76     typedef NaryFunctorImageFilter<TInputImage,TOutputImage,
77                                     Functor::ComposeRGBA<ITK_TYPENAME TInputImage::PixelType > >
78     Superclass;
79     typedef SmartPointer<Self> Pointer;
80     typedef SmartPointer<const Self> ConstPointer;
81
82     typedef typename Superclass::OutputImageType OutputImageType;
83
84     /** Method for creation through the object factory. */
85     itkNewMacro(Self);
86 protected:
87     ComposeRGBAImageFilter() {}
88     virtual ~ComposeRGBAImageFilter() {}
89 private:
90     ComposeRGBAImageFilter(const Self&); //purposely not implemented
91     void operator=(const Self&); //purposely not implemented
92
93
94
95 };
96
97 } // end namespace itk
98
99
100 #endif

```

Listing 1: itkComposeRGBAImageFilter.h.