

**Progress Report, February 19, 2010**

The aim of this project is to demonstrate a Biositemaps extension that permits the integration and interoperability of *Biositemaps*-described resources using graphical workflow environments. The basic prototype use-case of this extension uses Biositemaps.RDF/Biositemaps.XML formats and employs the LONI Pipeline as the resource integration environment. Other graphical workflow environments utilizing XML/RDF for module definition -- e.g., XIP, SCIRun, Kepler, Triana, Taverna, etc., will also be able to utilize extended versions of Biositemaps RDF objects developed and disseminated by NCBC, CTSA and other initiatives, research groups, institutions and individual investigators.

The proposed Biositemaps.RDF format extension allows the specification of 4 complementary descriptors to enhance resource computational invocation and tool-interoperability. These four types of descriptors include: execution, invocation, input/output/control tags. In addition, the extended RDF format allows for different (interpreter) resources to include or extract execution meta-data pertinent to their specific environment. For instance, graphical workflow environments like the LONI Pipeline, XIP, SCIRun, Kepler, Triana, Taverna, may share resource execution meta-data or have their own segments in the Biositemaps.RDF object. **Table 1** shows the core of the proposed Biositemaps.RDF extension.

**Table 1:** Proposed Biositemaps.RDF extension facilitating resource computational-invocation and tool-interoperability.

**Syntax of Biositemaps.RDF resource descriptions of execution, invocation, input/output and process control.**

```
<desc:invocation>
  <![CDATA[
    <loni_pipeline>
      Enter all appropriate LONI Pipeline execution/invocation/input/output/control syntax
    </loni_pipeline>
  ]]>
</desc:invocation>
```

The progress to-date is demonstrated with 3 examples of CCB Tools and resources included in the CCB Biositemaps object ([http://www.loni.ucla.edu/download/iTools/biositemap\\_NCBC\\_CCB.rdf](http://www.loni.ucla.edu/download/iTools/biositemap_NCBC_CCB.rdf)), which is registered online at <http://biositemaps.ncbcs.org/biositemap.registry>.

**Example 1 (LONI Shape-Distance)**

The LONI Shape-Distance is a program to calculate the distance between homologous vertices in a pair of Shapes. This program computes the distance between the vertices of two Shapes. The homology between vertices is based upon the order in which the vertices are added to the Shape's point sets; If you compare shapes A and B, the first distance value is that between vertex  $A_0$  and vertex  $B_0$ . Thus, the  $N^{\text{th}}$  distance value is that between vertices  $A_n$  and  $B_n$ . The distances so computed are written to an output file. Shape-Distance supports two output formats (version 1.0.0a0)

- a simple text file, with one distance per line, and
- a LONI UCF file, in which the distances are appended as a data attribute to the first shape given as an argument to the Shape-Distance application.

Two distance metrics are currently supported: Euclidean and Taxicab. The second shape file may contain more vertices than the first file. If this is true the extra vertices are ignored, unless the *-strict* flag is enabled, in which case the number of vertices in the two files must be equal. The current default output type is a text file. The file type is not automatically derived from the input files.

URL: <http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeDistance>

## Standard CCB Biositemaps RDF Shape-Distance RDF Description

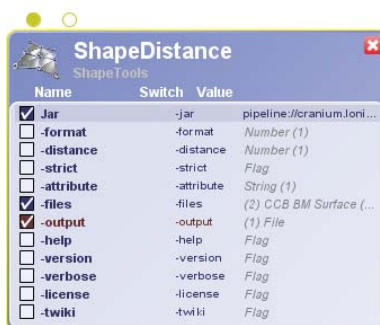
```
<desc:Resource_Description rdf:ID="LONI_ShapeDistance">
-
<desc:resource_type>
<BRO:Shape_Analysis/>
</desc:resource_type>
<desc:resource_name>LONI ShapeDistance</desc:resource_name>
-
<desc:version_information>
-
<desc:Version_Information>
<desc:version>1.0.0</desc:version>
<desc:release_date rdf:datatype="http://www.w3.org/2001/XMLSchema#string">10/31/2005</desc:release_date>
<desc:development_stage>Development with public userbase</desc:development_stage>
</desc:Version_Information>
</desc:version_information>
<desc:organization>UCLA Center for Computational Biology</desc:organization>
-
<desc:URL>
http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=26
</desc:URL>
-
<desc:keywords>
A tool for calculating the distance between vertices of two shape files
</desc:keywords>
<desc:license>CCB License</desc:license>
<desc:contact_person>Craig Schwartz</desc:contact_person>
-
<desc:description>
A tool for calculating the distance between vertices of two shape files
</desc:description>
</desc:Resource_Description>
```

## Pipeline required description of Shape-Distance

### • Pipeline module definition



### • Parameters



### • Executable

/usr/local/java/jdk1.6.0\_02/bin/java

## Extended CCB Biositemaps RDF Object

```
<desc:Resource_Description rdf:ID="LONI_ShapeDistance">
-
<desc:resource_type>
<BRO:Shape_Analysis/>
</desc:resource_type>
<desc:resource_name>LONI ShapeDistance</desc:resource_name>
-
<desc:version_information>
-
<desc:Version_Information>
<desc:version>1.0.0</desc:version>
<desc:release_date rdf:datatype="http://www.w3.org/2001/XMLSchema#string">10/31/2005</desc:release_date>
<desc:development_stage>Development with public userbase</desc:development_stage>
```

```

</desc:Version_Information>
</desc:version_information>
<desc:organization>UCLA Center for Computational Biology</desc:organization>
-
<desc:URL>
http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=26
</desc:URL>
-
<desc:keywords>
A tool for calculating the distance between vertices of two shape files
</desc:keywords>
<desc:license>CCB License</desc:license>
<desc:contact_person>Craig Schwartz</desc:contact_person>
-
<desc:description>
A tool for calculating the distance between vertices of two shape files
</desc:description>

<desc:invocation>
<![CDATA[
  <loni_pipeline>

<pipeline version="1">
  <moduleGroup id="ShapeTools_0" name="ShapeTools" description="Portable Geometry Tools" posX="0" posY="0">
    <module id="ShapeDistance_0" name="ShapeDistance" package="ShapeTools" version="1.1.0" description="A program to calculate the distance
between homologous vertices in a pair of Shapes.&#xA;&#xA;This program computes the distance between the vertices of two Shapes. The homology between
vertices is based upon the order in which the vertices are added to the Shape's point sets; If you compare shapes A and B, the first distance value is that between
vertex A_0 and vertex B_0. Thus, the Nth distance value is that between vertices A_n and B_n.&#xA;&#xA;The distances so computed are written to an output
file.&#xA;&#xA;Two output formats are supported (version 1.0.0a0) : a simple text file, with one distance per line, and a LONI Ucf file, in which the distances are
appended as a data attribute to the first shape given as an argument to the ShapeDistance application.&#xA;&#xA;Two distance metrics are currently supported:
Euclidean and Taxicab.&#xA;&#xA;The second shape file may contain more vertices than the first file. If this is true the extra vertices are ignored, unless the -
strict flag is enabled, in which case the number of vertices in the two files must be equal.&#xA;&#xA;Note The current default output type is a text file. The file
type is not automatically derived from the input files. " location="pipeline://cranium.loni.ucla.edu/usr/local/java/jdk1.6.0_02/bin/java" posX="150" posY="150"
executableVersion="1.0.0">
    <authors>
      <author
        fullName="ShapeTools
        Group"
        email="shapetls@loni.ucla.edu"
        website="http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeToolLibraryProgram" />
    </authors>
    <executableAuthors>
      <author
        fullName="ShapeTools
        Group"
        email="shapetls@loni.ucla.edu"
        website="http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeToolLibraryProgram" />
    </executableAuthors>
    <tag>shape</tag>
    <tag>distance</tag>
    <uri>http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeDistance</uri>
    <metadata>
      <data key="__creationDateKey" value="Tue May 19 11:04:35 PDT 2009" />
    </metadata>
    <input id="ShapeDistance_0.jar" name="jar" description="Application_Jar " enabled="true" required="true" order="0" switch="-jar"
switchSpaced="true">
      <format type="File" cardinality="1">
        <fileTypes>
          <filetype name="JAR" extension="jar" description="Sun JAR file" />
        </fileTypes>
      </format>
      <values>
        <value>pipeline://cranium.loni.ucla.edu/usr/local/ShapeTools-20090513/ShapeDistance.jar</value>
      </values>
    </input>
    <input id="ShapeDistance_0.-format" name="-format" description="Selects the format of the output shape." enabled="false" required="false"
order="1" switch="-format" switchSpaced="true">
      <format type="Number" cardinality="1" />
    </input>
    <input id="ShapeDistance_0.-distance" name="-distance" description="Selects the type of distance to measure." enabled="false" required="false"
order="2" switch="-distance" switchSpaced="true">
      <format type="Number" cardinality="1" />
    </input>
    <input id="ShapeDistance_0.-strict" name="-strict" description="Set to be true if number of vertices in both shapes must be equal." enabled="false"
required="false" order="3" switch="-strict" switchSpaced="true">
      <format type="String" cardinality="0" />
    </input>
    <input id="ShapeDistance_0.-attribute" name="-attribute" description="Sets the name of the distance attribute." enabled="false" required="false"
order="4" switch="-attribute" switchSpaced="true">
      <format type="String" cardinality="1" />
    </input>
    <input id="ShapeDistance_0.-files" name="-files" description="Two shape files whose inter-vertex distances will be calculated." enabled="true"
required="true" order="5" switch="-files" switchSpaced="true">
      <format type="File" cardinality="2">
        <fileTypes>
          <filetype name="CCB BM Surface" extension=".ccbzm" description="" />
          <filetype name="dfs" extension="dfs" description="" />
          <filetype name="DX file" extension="dx" description="A type of DX file." />
          <filetype name="MNI obj ascii" extension="obj" description="MNI obj ascii file" />
          <filetype name="PLY" extension="ply" description="" />
          <filetype name="UCF file" extension="ucf" description="" />
        </fileTypes>
      </format>
    </input>
  </moduleGroup>
</pipeline>
</![CDATA[
]]>

```

```

    </format>
  </input>
  <output id="ShapeDistance_0.-output" name="-output" description="The output file." enabled="true" required="true" order="6" switch="-output"
switchSpaced="true">
    <format type="File" cardinality="1">
      <fileTypes>
        <filetype name="File" extension="" description="Any type of data file" />
      </fileTypes>
    </format>
  </output>
  <input id="ShapeDistance_0.-help" name="-help" description="Prints help and usage information." enabled="false" required="false" order="7"
switch="-help" switchSpaced="true">
    <format type="String" cardinality="0" />
  </input>
  <input id="ShapeDistance_0.-version" name="-version" description="Prints version number of this application." enabled="false" required="false"
order="8" switch="-version" switchSpaced="true">
    <format type="String" cardinality="0" />
  </input>
  <input id="ShapeDistance_0.-verbose" name="-verbose" description="Enables detailed status messages." enabled="false" required="false"
order="9" switch="-verbose" switchSpaced="true">
    <format type="String" cardinality="0" />
  </input>
  <input id="ShapeDistance_0.-license" name="-license" description="Prints license details, and exits." enabled="false" required="false" order="10"
switch="-license" switchSpaced="true">
    <format type="String" cardinality="0" />
  </input>
  <input id="ShapeDistance_0.-twiki" name="-twiki" description="Writes command line arguments as a TWiki format table to output and exit."
enabled="false" required="false" order="11" switch="-twiki" switchSpaced="true">
    <format type="String" cardinality="0" />
  </input>
</module>
</moduleGroup>
</pipeline>

</loni_pipeline>
]]>
</desc:invocation>

</desc:Resource_Description>

```

## Example 2 (Shape-Transform)

The linear transform defined by a file is applied to the coordinates of the vertices in a shape. A copy of the input file is written with its vertex coordinates transformed into the output file. Supported transform filetypes are AIR Linear and ASCII matrices.

URL: <http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeTransform>

### Standard CCB Biositemaps RDF Shape-Distance RDF Description

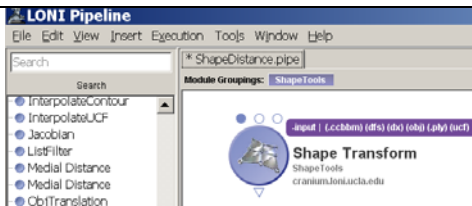
```

<desc:Resource_Description rdf:ID="ShapeTransform">
  <desc:keywords>The ShapeTransform application facilitates the moveof a shape file from one coordinate system to
another.</desc:keywords>
  <desc:description>The ShapeTransform application facilitates the moveof a shape file from one coordinate system to
another.</desc:description>
  <desc:resource_name>ShapeTransform</desc:resource_name>
  <desc:license>CCB License</desc:license>
  <desc:organization>UCLA Center for Computational Biology</desc:organization>
  <desc:version_information>
    <desc:Version_Information>
      <desc:release_date rdf:datatype="http://www.w3.org/2001/XMLSchema#string">5/3/2006</desc:release_date>
      <desc:version>1.1.0</desc:version>
      <desc:development_stage>Development with public userbase</desc:development_stage>
    </desc:Version_Information>
  </desc:version_information>
  <desc:URL>http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=30</desc:URL>
  <desc:contact_person>Bae Cheol Shin and Craig Schwartz</desc:contact_person>
<desc:resource_type>
  <BRO:Shape_Analysis/>
</desc:resource_type>
</desc:Resource_Description>

```

### Pipeline required description of Shape-Distance

- Pipeline module definition



- Parameters



- Executable

/usr/local/java/jdk1.6.0\_02/bin/java

### Extended CCB Biositemaps RDF Object

```

<desc:Resource_Description rdf:ID="ShapeTransform">
  <desc:keywords>The ShapeTransform application facilitates the moveof a shape file from one coordinate system to another.
</desc:keywords>
  <desc:description>The ShapeTransform application facilitates the moveof a shape file from one coordinate system to another.
</desc:description>
  <desc:resource_name>ShapeTransform</desc:resource_name>
  <desc:license>CCB License</desc:license>
  <desc:organization>UCLA Center for Computational Biology</desc:organization>
  <desc:version_information>
    <desc:Version_Information>
      <desc:release_date rdf:datatype="http://www.w3.org/2001/XMLSchema#string">5/3/2006</desc:release_date>
      <desc:version>1.1.0</desc:version>
      <desc:development_stage>Development with public userbase</desc:development_stage>
    </desc:Version_Information>
  </desc:version_information>
  <desc:URL>http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=30</desc:URL>
  <desc:contact_person>Bae Cheol Shin and Craig Schwartz</desc:contact_person>
  <desc:resource_type>
    <BRO:Shape_Analysis/>
  </desc:resource_type>

<desc:invocation>
  <![CDATA[
    <loni_pipeline>

<pipeline version=".1">
  <moduleGroup id="ShapeTools_0" name="ShapeTools" description="Portable Geometry Tools" posX="0" posY="0">
    <module id="ShapeTransform_0" name="Shape Transform" package="ShapeTools" version="1.1.0" description="The linear transform defined by a file
is applied to the coordinates of the vertices in a shape. A copy of the input file is written with it's vertex coordinates transformed into the output file. Supported
transform filetypes are AIR Linear and ascii matrix" location="pipeline://cranium.loni.ucla.edu/usr/local/java/jdk1.6.0_02/bin/java" posX="61" posY="45"
executableVersion="1.0.0">
      <authors>
        <author
          fullName="ShapeTools
          Group"
          email="shapetls@loni.ucla.edu"
          website="http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeToolLibraryProgram" />
      </authors>
      <executableAuthors>
        <author
          fullName="ShapeTools
          Group"
          email="shapetls@loni.ucla.edu"
          website="http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeToolLibraryProgram" />
      </executableAuthors>
      <tag>transform</tag>
      <uri>http://www.loni.ucla.edu/twiki/bin/view/CCB/ShapeTransform</uri>
      <metadata>
        <data key="__creationDateKey" value="Tue May 19 11:04:39 PDT 2009" />
      </metadata>
    </module>
  </moduleGroup>
  </pipeline>
  </desc:invocation>
  </desc:Resource_Description>
  
```

```

<input id="ShapeTransform_0.Jar" name="Jar" description="Application_Jar" enabled="true" required="true" order="0" switch="-jar"
switchSpaced="true">
  <format type="File" cardinality="1">
    <fileTypes>
      <filetype name="JAR" extension="jar" description="Sun JAR file" />
    </fileTypes>
  </format>
  <values>
    <value>pipeline://cranium.ionu.ucla.edu/usr/local/ShapeTools-20090513/ShapeTransform.jar</value>
  </values>
</input>
<input id="ShapeTransform_0.-xr" name="-xr" description="Sets rotation about the x axis, in degrees." enabled="false" required="false" order="1"
switch="-xr" switchSpaced="true">
  <format type="Number" cardinality="1" />
</input>
<input id="ShapeTransform_0.-yr" name="-yr" description="Sets rotation about the y axis, in degrees." enabled="false" required="false" order="2"
switch="-yr" switchSpaced="true">
  <format type="Number" cardinality="1" />
</input>
<input id="ShapeTransform_0.-zr" name="-zr" description="Sets rotation about the z axis, in degrees." enabled="false" required="false" order="3"
switch="-zr" switchSpaced="true">
  <format type="Number" cardinality="1" />
</input>
<input id="ShapeTransform_0.-xt" name="-xt" description="Sets translation on the x axis." enabled="false" required="false" order="4" switch="-xt"
switchSpaced="true">
  <format type="Number" cardinality="1" />
</input>
<input id="ShapeTransform_0.-yt" name="-yt" description="Sets translation on the y axis." enabled="false" required="false" order="5" switch="-yt"
switchSpaced="true">
  <format type="Number" cardinality="1" />
</input>
<input id="ShapeTransform_0.-zt" name="-zt" description="Sets translation on the z axis." enabled="false" required="false" order="6" switch="-zt"
switchSpaced="true">
  <format type="Number" cardinality="1" />
</input>
<input id="ShapeTransform_0.-air" name="-air" description="Read an AIR linear transform file." enabled="true" required="false" order="7"
switch="-air" switchSpaced="true">
  <format type="File" cardinality="1">
    <fileTypes>
      <filetype name="AIR file" extension="air" description="AIR Linear transformation" />
    </fileTypes>
  </format>
</input>
<input id="ShapeTransform_0.-ascii" name="-ascii" description="Read an ascii matrix linear transform file." enabled="false" required="false"
order="8" switch="-ascii" switchSpaced="true">
  <format type="File" cardinality="1">
    <fileTypes>
      <filetype name="File" extension="" description="Any type of data file" />
    </fileTypes>
  </format>
</input>
<input id="ShapeTransform_0.-inverse" name="-inverse" description="Apply inverse of specified transform." enabled="false" required="false"
order="9" switch="-inverse" switchSpaced="true">
  <format type="String" cardinality="0" />
</input>
<input id="ShapeTransform_0.-input" name="-input" description="the input file" enabled="true" required="true" order="10" switch="-input"
switchSpaced="true">
  <format type="File" cardinality="1">
    <fileTypes>
      <filetype name="CCB BM Surface" extension=".cbbm" description="" />
      <filetype name="dfs" extension="dfs" description="" />
      <filetype name="DX file" extension="dx" description="A type of DX file." />
      <filetype name="MNI obj ascii" extension="obj" description="MNI obj ascii file" />
      <filetype name="PLY" extension="ply" description="" />
      <filetype name="UCF file" extension="ucf" description="" />
    </fileTypes>
  </format>
</input>
<output id="ShapeTransform_0.-output" name="-output" description="the output file" enabled="true" required="true" order="11" switch="-output"
switchSpaced="true">
  <format type="File" cardinality="1">
    <fileTypes>
      <filetype name="File" extension="" description="Any type of data file" />
    </fileTypes>
  </format>
</output>
<input id="ShapeTransform_0.-examples" name="-examples" description="Prints examples of common usage." enabled="false" required="false"
order="12" switch="-examples" switchSpaced="true">
  <format type="String" cardinality="0" />
</input>
<input id="ShapeTransform_0.-help" name="-help" description="Prints help and usage information." enabled="false" required="false" order="13"
switch="-help" switchSpaced="true">
  <format type="String" cardinality="0" />
</input>
<input id="ShapeTransform_0.-version" name="-version" description="Prints version number of this application." enabled="false" required="false"
order="14" switch="-version" switchSpaced="true">
  <format type="String" cardinality="0" />

```

```

    </input>
    <input id="ShapeTransform_0.-verbose" name="-verbose" description="Enables detailed status messages." enabled="false" required="false"
order="15" switch="-verbose" switchSpaced="true">
      <format type="String" cardinality="0" />
    </input>
    <input id="ShapeTransform_0.-license" name="-license" description="Prints license details, and exits." enabled="false" required="false"
order="16" switch="-license" switchSpaced="true">
      <format type="String" cardinality="0" />
    </input>
    <input id="ShapeTransform_0.-twiki" name="-twiki" description="Writes command line arguments as a TWiki format table to output and exit."
enabled="false" required="false" order="17" switch="-twiki" switchSpaced="true">
      <format type="String" cardinality="0" />
    </input>
  </module>
</moduleGroup>
</pipeline>

  </loni_pipeline>
]]>
</desc:invocation>

</desc:Resource_Description>

```

### Example 3 (Wavelet Analysis of Image Registration, WAIR)

The WAIR software is a tool for quantitative analysis of various n-dimensional (n-D) image registration techniques. In particular, its applications include, but are not limited to, analyzing warp performance for stereotactic Human Brain anatomical and functional data. Using the Discrete Wavelet Transform (WT) we develop several image registration (warp) classification schemes. The first one, called "wavelet-space triangle analysis", is applicable for studying a family of warps on a single or multiple n-D data sets. For each data set the WAIR routine assigns a positive real number to every warp alignment in the family, and the best warp, for the given data, will be the one having the smallest value associated with it. The second classification method, called "cluster group classification", is applicable for analyzing the overall performance of a family of warps of a groups of data sets. Here, there is a single number assigned to each registration alignment, based on its group-clustering characteristics. Third goodness of warp approach, called SGC (spread group classification), is applicable for analyzing functional brain data. It gives preference to registration techniques that spread apart baseline versus activation functional signal for group data.

URL: [http://www.loni.ucla.edu/Software/Software\\_Detail.jsp?software\\_id=6](http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=6)

### Standard CCB Biositemaps RDF Shape-Distance RDF Description

```

<desc:Resource_Description rdf:ID="Wavelet_Analysis_of_Image_Registration">
-
<desc:resource_type>
<BRO:Wavelet_Transform/>
</desc:resource_type>
<desc:contact_person>Ivo D. Dinov</desc:contact_person>
<desc:URL>http://www.loni.ucla.edu/~dinov/WAIR.html</desc:URL>
<desc:organization>UCLA Center for Computational Biology</desc:organization>
-
<desc:description>
A tool for quantitative analysis of various n-dimensional (n-D) image registration techniques.
</desc:description>
<desc:license>CCB License</desc:license>
<desc:resource_name>Wavelet Analysis of Image Registration</desc:resource_name>
-
<desc:keywords>
A tool for quantitative analysis of various n-dimensional (n-D) image registration techniques.
</desc:keywords>
-
<desc:version_information>
-
<desc:Version_Information>
<desc:development_stage>Development with public userbase</desc:development_stage>
<desc:version>2</desc:version>
<desc:release_date rdf:datatype="http://www.w3.org/2001/XMLSchema#string">NA</desc:release_date>
</desc:Version_Information>
</desc:version_information>
</desc:Resource_Description>

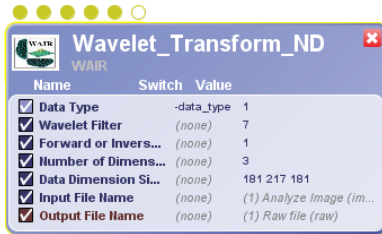
```

### Pipeline required description of Shape-Distance

- Pipeline module definition



- Parameters



- Executable

/usr/local/loniApps-1.0-4\_64bit/dinov/WT\_IWT

### Extended CCB Biositemaps RDF Object

```
<desc:Resource_Description rdf:ID="Wavelet_Analysis_of_Image_Registration">
-
<desc:resource_type>
<BRO:Wavelet_Transform/>
</desc:resource_type>
<desc:contact_person>Ivo D. Dinov</desc:contact_person>
<desc:URL>http://www.loni.ucla.edu/~dinov/WAIR.html</desc:URL>
<desc:organization>UCLA Center for Computational Biology</desc:organization>
-
<desc:description>
A tool for quantitative analysis of various n-dimensional (n-D) image registration techniques.
</desc:description>
<desc:license>CCB License</desc:license>
<desc:resource_name>Wavelet Analysis of Image Registration</desc:resource_name>
-
<desc:keywords>
A tool for quantitative analysis of various n-dimensional (n-D) image registration techniques.
</desc:keywords>
-
<desc:version_information>
-
<desc:Version_Information>
<desc:development_stage>Development with public userbase</desc:development_stage>
<desc:version>2</desc:version>
<desc:release_date rdf:datatype="http://www.w3.org/2001/XMLSchema#string">NA</desc:release_date>
</desc:Version_Information>
</desc:version_information>

<desc:invocation>
<![CDATA[
  <loni_pipeline>

<pipeline version=".1">
  <moduleGroup id="Untitled_0" name="Untitled" posX="0" posY="0">
    <module id="Wavelet_Transform_ND_0" name="Wavelet_Transform_ND" package="WAIR" version="2.0" description="The WAIR software is a tool for
quantitative analysis of various n-dimensional (n-D) image registration techniques. In particular, its applications include, but are not limited to, analyzing warp
performance for stereotactic Human Brain anatomical and functional data. Using the Discrete Wavelet Transform (WT) we develop several image registration
(warp) classification schemes. The first one, called wavelet-space triangle analysis, is applicable for studying a family of warps on a single or multiple n-D data
sets. For each data set the WAIR routine assigns a positive real number to every warp alignment in the family, and the best warp, for the given data, will be
the one having the smallest value associated with it. The second classification method, called cluster group classification, is applicable for analyzing the overall
performance of a family of warps of a groups of data sets. Here, there is a single number assigned to each registration alignment, based on its group-clustering
characteristics. Third goodness of warp approach, called SGC (spread group classification), is applicable for analyzing functional brain data. It gives preference
to registration techniques that spread apart baseline versus activation functional signal for group data."
location="pipeline://cranium.loni.ucla.edu/usr/local/loniApps-1.0-4_64bit/dinov/WT_IWT" posX="53" posY="81" executableVersion="2.0">
      <authors>
        <author fullName="Ivo Dinov" email="dinov@loni.ucla.edu" website="" />
      </authors>
      <executableAuthors>
        <author fullName="Ivo Dinov" email="dinov@loni.ucla.edu" website="" />
        <author fullName="Arthur W. Toga" email="toga@loni.ucla.edu" website="" />
      </executableAuthors>
      <tag>wavelets</tag>
      <tag>transform</tag>
      <tag>spectral</tag>
      <tag>registration</tag>
      <tag>evaluation</tag>
      <tag>volumetric</tag>
    </module>
  </moduleGroup>
</pipeline>
]]>
</desc:invocation>
```



```

<tag>frequency</tag>
<tag>space</tag>
<uri>http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=6</uri>
<metadata>
  <data key="__creationDateKey" value="Wed Mar 19 15:35:38 PDT 2008" />
</metadata>
<input id="Wavelet_Transform_ND_0.DataType" name="Data Type" description="Data Type: unsigned char (1Byte) or Float (4Byte):&#xA;4==Floating point Input (4Bytes), &#xA;1==1Byte Unsigned Char input" enabled="true" required="false" order="0" switch="-data_type" switchSpaced="true">
  <format type="Number" cardinality="1" />
  <values>
    <value>1</value>
  </values>
</input>
<input id="Wavelet_Transform_ND_0.WaveletFilter" name="Wavelet Filter" description="Wavelet Filter. See: http://www.loni.ucla.edu/~dinov/WAIR.dir/WAIR_Usage.pdf&#xA;WaveletBasis==1 , Spline_3_7; WaveletBasis==2 , Spline_3_3; WaveletBasis==3 , Spline_2_4; WaveletBasis==4 , Spline_2_2; WaveletBasis==5 , Pseudocoiflet_4_4; WaveletBasis==6 , Haar; WaveletBasis==7 , Daubechies_20; WaveletBasis==8 , Daubechies_12; WaveletBasis==9 , Daubechies_10; WaveletBasis==10 , Daubechies_8; WaveletBasis==11 , Daubechies_6; WaveletBasis==12 , Daubechies_4; WaveletBasis==13 , Coiflet_6; WaveletBasis==14 , Coiflet_4; WaveletBasis==15 , Coiflet_2; WaveletBasis==16 ,BurtAdelson;&#xA;WaveletBasis==17 , BattleLemarie;" enabled="true" required="true" order="1">
  <format type="Number" cardinality="1" />
  <values>
    <value>7</value>
  </values>
</input>
<input id="Wavelet_Transform_ND_0.ForwardorInverseWaveletTransform" name="Forward or Inverse Wavelet Transform" description="Which Transform Choose: 1 for (forward) WT, or 0; for (inverse) IWT" enabled="true" required="true" order="2">
  <format type="Number" cardinality="1" />
  <values>
    <value>1</value>
  </values>
</input>
<input id="Wavelet_Transform_ND_0.NumberofDimensions" name="Number of Dimensions" description="Choose the Number of dimensions&#xA;Num_Dim = integer representing the number of dimensions of the data set" enabled="true" required="true" order="3">
  <format type="Number" cardinality="1" />
  <values>
    <value>3</value>
  </values>
</input>
<input id="Wavelet_Transform_ND_0.DataDimensionSizes" name="Data Dimension Sizes" description="Dimension Sizes. Enter a string of space-separated Integers representing the size of the first (fastest varying index), second, third, ..., last-dimension-size (slowest-varying index:&#xA;x_1_Dim_Size x_2_Dim_Size ... x_N_Dim_Size" enabled="true" required="true" order="4">
  <format type="String" cardinality="1" />
  <values>
    <value>181 217 181</value>
  </values>
</input>
<input id="Wavelet_Transform_ND_0.InputFileName" name="Input File Name" description="Input File Name (Raw data)" enabled="true" required="true" order="5">
  <format type="File" cardinality="1">
    <fileTypes>
      <filetype name="Analyze Image" extension="img" description="Analyze Image">
        <need>hdr</need>
      </filetype>
    </fileTypes>
  </format>
</input>
<output id="Wavelet_Transform_ND_0.OutputFileName" name="Output File Name" description="Name of the output file that will contain the floating-point array of the WT of the input data" enabled="true" required="true" order="6">
  <format type="File" cardinality="1">
    <fileTypes>
      <filetype name="Raw file" extension="raw" description="" />
    </fileTypes>
  </format>
</output>
</module>
</moduleGroup>
</pipeline>

</loni_pipeline>
]]>
</desc:invocation>

</desc:Resource_Description>

```

## Summary

The Pipeline environment can now load and interpret the extended CCB Biositemaps Object (**biositemap\_NCBC\_CCB\_PL\_update021610.rdf**) and instantiate the 3 resources (Shape-Distance, Shape-Transform, and the WAIR n-D Wavelet Transform) directly from the Biositemaps.RDF file.

**Include a Pipeline image capturing this demonstration!**

**Supporting Docs:**

- Original CCB Biositemaps RDF doc: [www.loni.ucla.edu/download/iTools/biositemap\\_NCBC\\_CCB.rdf](http://www.loni.ucla.edu/download/iTools/biositemap_NCBC_CCB.rdf).
- The Pipeline XML syntax/language is described online at: <http://pipeline.loni.ucla.edu/support/xml-overview/>.
- Validate the CCB Biositemaps RDF doc containing the Pipeline XML-descriptions for the 3 tools above (biositemap\_NCBC\_CCB\_PL\_update021610.rdf).
- Three \*.pipe files for the 3 CCB Resources (ShapeDistance.pipe, ShapeTransform.pipe, ND\_WaveletTransform\_WAIR.pipe).