SentiSculpt 1.0 SDK

Developer’s Guide
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>2</td>
</tr>
<tr>
<td>About</td>
<td>2</td>
</tr>
<tr>
<td>Supported Platforms</td>
<td>2</td>
</tr>
<tr>
<td>System Requirements</td>
<td>2</td>
</tr>
<tr>
<td>Whats new</td>
<td>2</td>
</tr>
<tr>
<td>Overview</td>
<td>3</td>
</tr>
<tr>
<td>System Components</td>
<td>3</td>
</tr>
<tr>
<td>StructureFromMotion</td>
<td>4</td>
</tr>
<tr>
<td>Mesher</td>
<td>5</td>
</tr>
<tr>
<td>Texturer</td>
<td>6</td>
</tr>
<tr>
<td>Using the point cloud data object</td>
<td>6</td>
</tr>
<tr>
<td>Samples</td>
<td>6</td>
</tr>
<tr>
<td>MeshLab Plugin</td>
<td>7</td>
</tr>
<tr>
<td>Photo shooting guide line</td>
<td>9</td>
</tr>
<tr>
<td>General information about cameras</td>
<td>9</td>
</tr>
<tr>
<td>General information about objects and scenes</td>
<td>10</td>
</tr>
<tr>
<td>Shooting an object</td>
<td>10</td>
</tr>
<tr>
<td>Shooting an exterior</td>
<td>13</td>
</tr>
<tr>
<td>Shooting an interior</td>
<td>13</td>
</tr>
<tr>
<td>Summary</td>
<td>14</td>
</tr>
<tr>
<td>API reference</td>
<td>16</td>
</tr>
<tr>
<td>C Reference</td>
<td>16</td>
</tr>
<tr>
<td>NSfmDensePointGenerationOptions.h File Reference</td>
<td>16</td>
</tr>
<tr>
<td>NSfmMatrix.h File Reference</td>
<td>23</td>
</tr>
<tr>
<td>NSfmMesher.h File Reference</td>
<td>24</td>
</tr>
<tr>
<td>NSfmPointCloud.h File Reference</td>
<td>26</td>
</tr>
<tr>
<td>NSfmTaskProgress.h File Reference</td>
<td>32</td>
</tr>
<tr>
<td>NSfmTypes.h File Reference</td>
<td>33</td>
</tr>
<tr>
<td>NStructureFromMotion.h File Reference</td>
<td>36</td>
</tr>
<tr>
<td>.Net Reference</td>
<td>42</td>
</tr>
<tr>
<td>Namespace List</td>
<td>42</td>
</tr>
<tr>
<td>Class List</td>
<td>42</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.NSfmDensePointGenerationOptions Class Reference</td>
<td>43</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.Exceptions.NSfmException Class Reference</td>
<td>47</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.NSfmMatrix&lt;T&gt; Class Template Reference</td>
<td>48</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.NSfmMesher Class Reference</td>
<td>48</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.NSfmObject Class Reference</td>
<td>50</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.NSfmPointCloud Class Reference</td>
<td>50</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.NSfmTaskProgress Class Reference</td>
<td>55</td>
</tr>
<tr>
<td>Neurotec.SentiSculpt.NStructureFromMotion Class Reference</td>
<td>56</td>
</tr>
</tbody>
</table>
**Introduction**

**About**
SentiSculpt is an SDK for structure from motion. It can generate 3D structural information as a point cloud from a range of input images. Furthermore, it can be used to generate meshes from such point clouds and perform texture mapping.

**Supported Platforms**
The table below shows relations between different operating systems and processors:

<table>
<thead>
<tr>
<th>Processor architecture</th>
<th>MS Windows</th>
</tr>
</thead>
<tbody>
<tr>
<td>x86_64 (x64-64)</td>
<td>+</td>
</tr>
</tbody>
</table>

**System Requirements**
Operating System requirements
- Microsoft Windows XP/2003/2008/Vista/7/8* 64-bit platform.
- Microsoft .NET framework 3.5 client profile or newer (for .NET components usage).

Hardware requirements
- PC with x86-64 (64-bit) compatible processor. 2GHz or better processor is recommended.
- At least 1 GB of free RAM is recommended for the application. However, memory requirements will vary depending on input image size and quality. Dense reconstruction will require more RAM and it depends on the dense reconstruction tool being used.
- Video card supporting OpenGL 1.5 or later.

Imaging recommendations
- Image resolution 2~5 megapixels
- 20-80 input Images

**Whats new**
This section covers change log, fixes and new features added to different versions of SentiSculpt.

February 23, 2015
- First Release V 1.0.0.2

February 25, 2015 (V 1.0.0.3)
- Added C API sample

March 04, 2015 (V 1.0.0.4)
- Added PMVS2 options to MeshLab sample
March 10, 2015 (V 1.0.0.5)
▪ Added support to export visibility information

April 20, 2015 (V 1.0.0.6)
▪ Improved TextureMapper.exe error handling

November 17, 2016 (V 1.0.1.7)
▪ Upgraded SDK components to support new licensing model

May 20, 2016 (V 1.0.2.8)
▪ Improved error messages in samples and MeshLab plugin.
▪ Upgraded SDK components

**Overview**

SentiSculpt is a Software Development Kit (SDK) for creation of 3D point clouds and meshes. It can also perform texture mapping around meshes it generates.

**System Components**

SentiSculpt consist of three main components, namely StructuerFromMotion, Mesher and Texturer. Figure 1.1 shows a graphical representation of these components and their workflow.
**StructuerFromMotion**
Generates a sparse point cloud from a given set of images. It generates a point cloud data object as the output. Point cloud data object contains generated 3D points as well as various meta-data such as camera coordinates and the set of images used to generate those 3D points. It should be noted that only a sparse point cloud is generated by this component. Generated sparse point cloud can be used to generate a dense point cloud using external dense point cloud generation tools, such as PMVS2. Once the dense point cloud is generated, it should be imported back to the point cloud generated by this module. Refer SDK samples for more details. Figure 1.2 gives a detail view of the module.

<table>
<thead>
<tr>
<th>Input data</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Images</td>
<td>JPG</td>
</tr>
</tbody>
</table>
Fig 1.2. Detail view of StructureFromMotion generator

**Mesher**

Performs mesh reconstruction using points saved in a point cloud data object and saves the generated one or more meshes back in to the same point cloud data object. Mesher can also load points directly from PLY files.

Fig 1.3. Detail view of Mesher
Texturer
Applies texture to meshes saved in the point cloud data object. It generates one or more texture mapped meshes and their relevant texture images. This is an external executable and refer sample projects how to execute it.

Fig 1.4. Detail view of Texturer

Using the point cloud data object
Point cloud data object is a common data buffer used by SentiScuplt components. It contains output from structure from motion generator, mesher and texture. There are readers and writers to export and import those data.

<table>
<thead>
<tr>
<th>Import</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense point list (form file)</td>
<td>PLY</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Export</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense point list</td>
<td>PLY, ASC</td>
</tr>
<tr>
<td>Sparse point list</td>
<td>PLY, ASC</td>
</tr>
<tr>
<td>Base Image list used to generate 3D points</td>
<td>JPG</td>
</tr>
<tr>
<td>Calculated Projection Matrices</td>
<td>TXT</td>
</tr>
<tr>
<td>Calculated Camera coordinates</td>
<td>TXT</td>
</tr>
<tr>
<td>Reconstructed Mesh</td>
<td>OBJ</td>
</tr>
</tbody>
</table>

Samples
Sample programs to demonstrate SDK can be found in API/Samples folder. Following code snippets demonstrate performing critical API functions in C++.
▪ Creating the sparse point list and obtaining the generated point cloud

```cpp
NStructureFromMotion sfm;
SfmTaskResult result = sfm.NSfmGeneratePointCloudFromFolder(folderPath.c_str());
if (result == Success)
{
    pointCloud = sfm.GetPointCloud();
}
```

▪ Loading a dense point list to the point cloud. Refer SDK samples how to execute such tools programmatically. Note that dense point cloud should be in “.PLY” format.

```cpp
pointCloud->ImportDensePointsFromFile(plyPath.c_str());
```

▪ Mesh reconstruction

```cpp
NSfmMesher mesher;
mesher.ReconstructFromPointCloud(Neurotec::SentiSculpt::Poisson, High, *pointCloud);
```

▪ Once mesh reconstruction is finished, it can be exported for texture mapping. Refer SDK samples how to execute texture mapping tool.

### MeshLab Plugin

SentiSight SDK contains a MeshLab executable with a plugin capable of loading SentiSculpt modules. Users can open a text file containing names of a set of JPG images and the plugin generates texture mapped mesh.

Go to “Import Mesh” option under the “File” menu and select ”Image list for SentiSculpt”, as shown in figure 3.1.
Fig. 3.1. Importing an image list for SentiSculpt plugin.

Plugin will prompt a parameter configuration window, where user can change various parameters used for point cloud generation. Refer API specification for more details on configuring general parameters and it is possible to continue with the default parameters assigned in the parameter window. Please read PMVS2 documentation for more information on PMVS2 options.

Fig. 3.2. Parameter window showing the list of parameters with their default values.

Figure 3.3 shows the output for Girl_with_a_Dog sample dataset provided with the SentiSculpt SDK.
Fig. 3.3. MeshLab plugin output using Girl_with_a_Dog image set.

Photo shooting guide line

General information about cameras
SentiSculpt can be used with different types of cameras starting from the smart phones and ending with a highly professional DSLR cameras. The quality of the reconstruction will depend on the quality of the input images. So cameras with less distortions and more qualitative imaging subsystem will likely produce better results.
During shooting sessions some basic rules should be preserved.
• **Constant focal length** during the whole session - **do not use zoom in/out** after you started the session. Pre-adjust camera's zoom before starting acquisition process. Later, if you need more details you can go closer to the object.

• Try to ensure **constant lighting conditions** - varying lighting conditions during the same session can lead to an inaccurate results, or even to a failure.

• **Disable automatic adjustments** on your camera - most of the cameras have auto focusing enabled.

• **Disable flash light** - flash light can be the source of high reflections.

• **Do not use digital zooming** - digital zooming leads to all kinds of distortions and noise.

After shooting session is done, **do not use any kind of post-process** on images. If for some reasons you have to edit the pictures, for example, resize or crop make sure that all pictures of the same session will be processed using the same settings.

There is no limitation on resolution of the images, but keep in mind that processing of high resolution images is time consuming operation. Images of 5 megapixel should give a good results. Finally **avoid blurry images** you can use tripod.

**General information about objects and scenes**

To get a good reconstruction results something between 20-80 images of the object should be used. Every feature of the object should be visible in at least 3 adjacent images. Adjacent images must overlap approximately by 2/3. The best way to capture the whole object is by taking images in circular trajectory using angle of rotation about 15 degrees.

There are some limitations to the objects and scenes. Not all objects can be reconstructed properly. SentiSculpt is used to reconstruct static scenes, so it will mostly fail while trying to reconstruct some moving parts. Anyway some occlusions are allowed even if they are moving. Another thing which is very hard to reconstruct is transparent objects or their parts. So glasses, windows or water probably will not be reconstructed. Shiny or reflective surfaces should be avoided as well. Finally you should avoid un-textured objects like plain walls or periodically textured objects like chess board (if it covers the most part of the image).

**Shooting an object**

While shooting an object please keep in mind that algorithm is looking for common features at least in three images, unless other non-default settings are used. One of the main rules while making a photo session is to make adjacent images overlap approximately by 2/3 (Fig. 4.1.)
To achieve the best reconstruction results we recommend to make a circular trajectory around the object using angle of rotation about 15 degrees while capturing it. So that makes about 20-30 images of the object in one round (Fig. 4.2.).

In most cases some parts of the object will be occluded, or not seen by the camera from its point of view. To avoid that, object can be captured using different levels of the camera elevation. For example one level slightly from the bottom, another level facing object directly and one level slightly from the top (Fig. 4.3.). Just remember the rule of 2/3. It should be maintained in vertical direction as well as in horizontal.
After you have finished capturing the whole object, maybe, using different levels of elevation, then you can proceed with capturing some details. **Do not use zooming function of your camera.** Just move closer to the object. While moving make couple of additional shots (Fig. 4.4.) so that the size of the object would not change drastically, otherwise it can be hard to resolve geometry of the object and the whole reconstruction can fail.
Figure 4.5 shows how the shooting session might look like. First capture the whole object, then go with the details. For successful reconstruction number of images should be in range between 20 and 80 depending on the level of the details that you want to reconstruct. We give no limitation neither on the number of the images neither on their resolution. But keep in mind that reconstructing the object from more images or using high resolution photos not always gives better results not to mention that reconstruction time in these cases increase drastically and in some cases this even can lead to the failure of the whole reconstruction process.

**Shooting an exterior**

Shooting an exterior can be a challenging task. For example houses or facades are quite large objects and it is hard to capture them. Often you will have to move back to fit this kind of the objects in the camera's image. But you do not have to fit the whole object in the image, reconstruction can be done even from partial views of the object. Just once again remember the rule of 2/3. Make overlapping images. While capturing the house of the facade try to move across trajectory parallel to the walls. Take a special care in the area near the corners, these are the hardest. Try to think of specific shooting order to avoid missing parts and make sure that common features are seen in at least 3 adjacent views (Fig. 4.6.).

![Fig. 4.6. Taking images of the house or facade.](image)

**Shooting an interior**

Shooting interior is as challenging as the exterior. It this case the opposite approach should be used. Try standing so that your back is facing the wall and capture the opposite side of the room moving in a circular trajectory parallel to the walls (Fig. 4.7.). Once again remember the rule of 2/3 and make overlapping images so that common features are seen in at least 3 adjacent views.
The most common mistakes that can be done while capturing the room is shooting the walls while standing in front of them at close range (Fig. 4.8. left) or trying to capture the whole room while standing in the middle of it and just rotating the camera (Fig. 4.8. right). In these cases it is hard to resolve the geometry and the reconstruction will mostly fail. Another thing which often leads to failure is the lack of features (textures) in the room. Most of the walls are painted in homogeneous light colors.

**Summary**

Make sure following guidelines are followed.

- **Camera setup**
  - Constant focal length
  - No zooming in/out between shots of the same session
  - No digital zoom
  - Use tripod where possible
- **Image capturing**
- Avoid blurry images
- Avoid images with shiny, transparent un-textures or periodically textured objects or their parts
- Images must overlap approximately by 2/3
- Same features of the object should be visible at least in 3 adjacent photos
- Capture images of the object in circular trajectory using angle of rotation about 15 degrees
API reference

C Reference

NSfmDensePointGenerationOptions.h File Reference

Macros
#define SFM_NSFM_DENSE_POINT_GENERAITON_OPTIONS_H

Functions
NResult N_API NSfmDensePointGenerationOptionsCreate (HNSfmDensePointGenerationOptions phOptions)
NSfmDensePointGenerationOptionsCreate Initializes NSfmDensePointGenerationOptionsImpl instance.

NResult N_API NSfmDensePointGenerationOptionsDestroy (HNSfmDensePointGenerationOptions phOptions)
NSfmDensePointGenerationOptionsDestroy Destroys the HNSfmDensePointGenerationOptions instance.

NResult N_API NSfmDensePointGenerationOptionsGetLevel (HNSfmDensePointGenerationOptions phOptions, int *level)
NSfmDensePointGenerationOptionsGetLevel Returns option level.

NResult N_API NSfmDensePointGenerationOptionsSetLevel (HNSfmDensePointGenerationOptions phOptions, int level)
NSfmDensePointGenerationOptionsSetLevel Sets option level.

NResult N_API NSfmDensePointGenerationOptionsGetCsize (HNSfmDensePointGenerationOptions phOptions, int *csize)
NSfmDensePointGenerationOptionsGetCsize Returns option csie.

NResult N_API NSfmDensePointGenerationOptionsSetCsize (HNSfmDensePointGenerationOptions phOptions, int csize)
NSfmDensePointGenerationOptionsSetCsize Sets option csize.

NResult N_API NSfmDensePointGenerationOptionsGetThreshold (HNSfmDensePointGenerationOptions phOptions, float *threshold)
NSfmDensePointGenerationOptionsGetThreshold Returns option threshold.

NResult N_API NSfmDensePointGenerationOptionsSetThreshold (HNSfmDensePointGenerationOptions phOptions, float threshold)
NSfmDensePointGenerationOptionsSetThreshold Sets option threshold.

NResult N_API NSfmDensePointGenerationOptionsGetWsize (HNSfmDensePointGenerationOptions phOptions, int *wsize)
NSfmDensePointGenerationOptionsGetWsize Returns option wsize.

NResult N_API NSfmDensePointGenerationOptionsSetWsize (HNSfmDensePointGenerationOptions phOptions, int wsize)
NSfmDensePointGenerationOptionsSetWsize Sets option wsize.

NResult N_API NSfmDensePointGenerationOptionsGetMinImageNum (HNSfmDensePointGenerationOptions phOptions, int *minImageNum)
NSfmDensePointGenerationOptionsGetMinImageNum Returns option minImageNum.

NResult N_API NSfmDensePointGenerationOptionsSetMinImageNum (HNSfmDensePointGenerationOptions phOptions, int minImageNum)
NSfmDensePointGenerationOptionsSetMinImageNum Sets option minImageNum.

NResult N_API NSfmDensePointGenerationOptionsGetCpu (HNSfmDensePointGenerationOptions phOptions, int *cpu)
NSfmDensePointGenerationOptionsGetCpu Returns option cpu.
NResult N_API NSfmDensePointGenerationOptionsSetCpu (HNSfmDensePointGenerationOptions phOptions, int cpu)  
NSfmDensePointGenerationOptionsSetCpu Sets option cpu.

NResult N_API NSfmDensePointGenerationOptionsGetQuad (HNSfmDensePointGenerationOptions phOptions, float *quad)  
NSfmDensePointGenerationOptionsGetQuad Returns option quad.

NResult N_API NSfmDensePointGenerationOptionsSetQuad (HNSfmDensePointGenerationOptions phOptions, float quad)  
NSfmDensePointGenerationOptionsSetQuad Sets option quad.

NResult N_API NSfmDensePointGenerationOptionsGetMaxAngle (HNSfmDensePointGenerationOptions phOptions, int *angle)  
NSfmDensePointGenerationOptionsGetMaxAngle Returns option max angle.

NResult N_API NSfmDensePointGenerationOptionsSetMaxAngle (HNSfmDensePointGenerationOptions phOptions, int angle)  
NSfmDensePointGenerationOptionsSetMaxAngle Sets option angle.

NResult N_API NSfmDensePointGenerationOptionsGetUseVisData (HNSfmDensePointGenerationOptions phOptions, int *useVisData)  
NSfmDensePointGenerationOptionsGetUseVisData Returns option useVisData.

NResult N_API NSfmDensePointGenerationOptionsSetUseVisData (HNSfmDensePointGenerationOptions phOptions, int useVisData)  
NSfmDensePointGenerationOptionsSetUseVisData Sets option useVisData.

NResult N_API NSfmDensePointGenerationOptionsGetSequence (HNSfmDensePointGenerationOptions phOptions, int *sequence)  
NSfmDensePointGenerationOptionsGetSequence Returns option sequence.

NResult N_API NSfmDensePointGenerationOptionsSetSequence (HNSfmDensePointGenerationOptions phOptions, int sequence)  
NSfmDensePointGenerationOptionsSetSequence Sets option sequence.

---

**Detailed Description**

File containing dense point cloud generation options.

**Author:**
Neurotechnology

**Date:**
09 February 2015

---

**Macro Definition Documentation**

```
#define SFM_NSFM_DENSE_POINT_GENERATION_OPTIONS_H
```

---

**Function Documentation**

NResult N_API NSfmDensePointGenerationOptionsCreate (HNSfmDensePointGenerationOptions *phOptions)

NSfmDensePointGenerationOptionsCreate Initializes NSfmDensePointGenerationOptionsImpl instance.

**Returns:**

NResult Indicates success or failures
Parameters:

| phOptions       | Pointer to be initialized |

NResult N_API NSfmDensePointGenerationOptionsDestroy (HNSfmDensePointGenerationOptions phOptions)

NSfmDensePointGenerationOptionsDestroy Destroys the HNSfmDensePointGenerationOptions instance.

Returns:
NResult Indicates success or failures

Parameters:

| phOptions       | Pointer to the instance to be destroyed. |

NResult N_API NSfmDensePointGenerationOptionsGetCpu (HNSfmDensePointGenerationOptions phOptions, int * cpu)

NSfmDensePointGenerationOptionsGetCpu Returns option cpu.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int*</td>
<td>value assigned for option cpu</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsGetCsize (HNSfmDensePointGenerationOptions phOptions, int * csize)

NSfmDensePointGenerationOptionsGetCsize Returns option csize.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int*</td>
<td>value assigned for option csize</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsGetLevel (HNSfmDensePointGenerationOptions phOptions, int * level)

NSfmDensePointGenerationOptionsGetLevel Returns option level.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int*</td>
<td>value assigned for option level</td>
</tr>
</tbody>
</table>
NResult N_API NSfmDensePointGenerationOptionsGetMinImageNum
(HNSfmDensePointGenerationOptions phOptions, int * minImageNum)

NSfmDensePointGenerationOptionsGetMinImageNum Returns option minImageNum.

Returns:
    NResult Indicates success or failures
Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int*</td>
<td>value assigned for option minImageNum</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsGetSequence
(HNSfmDensePointGenerationOptions phOptions, int * sequence)

NSfmDensePointGenerationOptionsGetSequence option sequence.

Returns:
    NResult Indicates success or failures
Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int*</td>
<td>value assigned for option sequence</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsGetQuad
(HNSfmDensePointGenerationOptions phOptions, float * quad)

NSfmDensePointGenerationOptionsGetQuad Returns option quad.

Returns:
    NResult Indicates success or failures
Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>float *</td>
<td>value assigned for option quad</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsGetThreshold
(HNSfmDensePointGenerationOptions phOptions, float * threshold)

NSfmDensePointGenerationOptionsGetThreshold Returns option threshold.

Returns:
    NResult Indicates success or failures
Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int*</td>
<td>value assigned for option threshold</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsGetMaxAngle
(HNSfmDensePointGenerationOptions phOptions, int * angle)
**NSfmDensePointGenerationOptionsGetMaxAngle** Returns option max angle.

**Returns:**
NResult Indicates success or failures

**Parameters:**
- `phOptions` HNSfmDensePointGenerationOptions instance
- `int*` value assigned for option max angle

**NResult N_API NSfmDensePointGenerationOptionsGetUseVisData**
(HNSfmDensePointGenerationOptions `phOptions`, int * `useVisData`)

NSfmDensePointGenerationOptionsGetUseVisData Returns option useVisData.

**Returns:**
NResult Indicates success or failures

**Parameters:**
- `phOptions` HNSfmDensePointGenerationOptions instance
- `int*` value assigned for option useVisData

**NResult N_API NSfmDensePointGenerationOptionsGetWsize**
(HNSfmDensePointGenerationOptions `phOptions`, int * `wsize`)

NSfmDensePointGenerationOptionsGetWsize Returns option wsize.

**Returns:**
NResult Indicates success or failures

**Parameters:**
- `phOptions` HNSfmDensePointGenerationOptions instance
- `int*` value assigned for option wsize

**NResult N_API NSfmDensePointGenerationOptionsSetCpu** (HNSfmDensePointGenerationOptions `phOptions`, int `cpu`)

NSfmDensePointGenerationOptionsSetCpu Sets option cpu.

**Returns:**
NResult Indicates success or failures

**Parameters:**
- `phOptions` HNSfmDensePointGenerationOptions instance
- `int` value to be assigned for option cpu

**NResult N_API NSfmDensePointGenerationOptionsSetCsize**
(HNSfmDensePointGenerationOptions `phOptions`, int `csize`)

NSfmDensePointGenerationOptionsSetCsize Sets option csize.
Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option csize</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsSetLevel
(HNSfmDensePointGenerationOptions phOptions, int level)

NSfmDensePointGenerationOptionsSetLevel Sets option level.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option level</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsSetMinImageNum
(HNSfmDensePointGenerationOptions phOptions, int minImageNum)

NSfmDensePointGenerationOptionsSetMinImageNum Sets option minImageNum.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option minImageNum</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsSetSequence
(HNSfmDensePointGenerationOptions phOptions, int sequence)

NSfmDensePointGenerationOptionsSetSequence Sets option sequence.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option sequence</td>
</tr>
</tbody>
</table>

NResult N_API NSfmDensePointGenerationOptionsSetQuad
(HNSfmDensePointGenerationOptions phOptions, float quad)

NSfmDensePointGenerationOptionsSetQuad Sets option quad.

Returns:
NResult Indicates success or failures
Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>float</td>
<td>value to be assigned for option quad</td>
</tr>
</tbody>
</table>

NResult N_API NSmfDensePointGenerationOptionsSetThreshold
(HNSfmDensePointGenerationOptions phOptions, float threshold)

NSfmDensePointGenerationOptionsSetThreshold Sets option threshold.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option threshold</td>
</tr>
</tbody>
</table>

NResult N_API NSmfDensePointGenerationOptionsSetMaxAngle
(HNSfmDensePointGenerationOptions phOptions, int angle)

NSfmDensePointGenerationOptionsSetAngle Sets option angle.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option max angle</td>
</tr>
</tbody>
</table>

NResult N_API NSmfDensePointGenerationOptionsSetUseVisData
(HNSfmDensePointGenerationOptions phOptions, int useVisData)

NSfmDensePointGenerationOptionsSetUseVisData Sets option useVisData.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option useVisData</td>
</tr>
</tbody>
</table>

NResult N_API NSmfDensePointGenerationOptionsSetWsize
(HNSfmDensePointGenerationOptions phOptions, int wsize)

NSfmDensePointGenerationOptionsSetWsize Sets option wsize.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phOptions</th>
<th>HNSfmDensePointGenerationOptions instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option wsize</td>
</tr>
</tbody>
</table>
NSfmMatrix.h File Reference

Macros
#define SFM_NSFM_MATRIX_H

Functions
NResult N_API NSfmMatrixCreate_D (HNSfmMatrixD *phMatrix, int rowCount, int columnCount)

NSfmMatrixCreate_D Initializes HNSfmMatrixD instance.

NResult N_API NSfmMatrixDestroy_D (HNSfmMatrixD phMatrix)

NSfmMatrixDestroy_D Destroys the HNSfmMatrixD instance.

NResult N_API NSfmMatrixGetRowCount_D (HNSfmMatrixD phMatrix, int *rowCount)

NSfmMatrixGetRowCount_D Returns row count in the matrix.

NResult N_API NSfmMatrixGetColumnCount_D (HNSfmMatrixD phMatrix, int *columnCount)

NSfmMatrixGetColumnCount_D Returns column count in the matrix.

NResult N_API NSfmMatrixGetAt_D (HNSfmMatrixD phMatrix, int row, int column, double *value)

NSfmMatrixGetAt_D Returns value at the given position.

Macro Definition Documentation
#define SFM_NSFM_MATRIX_H

Function Documentation
NResult N_API NSfmMatrixCreate_D (HNSfmMatrixD * phMatrix, int rowCount, int columnCount)

NSfmMatrixCreate_D Initializes HNSfmMatrixD instance.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phMatrix</th>
<th>Pointer to be initialized</th>
</tr>
</thead>
<tbody>
<tr>
<td>rowCount</td>
<td>Number of rows in the matrix</td>
</tr>
<tr>
<td>columnCount</td>
<td>Number of columns in the matrix</td>
</tr>
</tbody>
</table>

NResult N_API NSfmMatrixDestroy_D (HNSfmMatrixD phMatrix)

NSfmMatrixDestroy_D Destroys the HNSfmMatrixD instance.

Returns:
NResult Indicates success or failures

Parameters:

| phMatrix | Pointer to the instance to be destroyed. |

NResult N_API NSfmMatrixGetAt_D (HNSfmMatrixD phMatrix, int row, int column, double * value)
NSfmMatrixGetAt_D Returns value at the given position.

Returns:
NResult

Parameters:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>phMatrix</td>
<td>HNSfmMatrixD instance</td>
</tr>
<tr>
<td>row</td>
<td>Row number</td>
</tr>
<tr>
<td>column</td>
<td>Column number</td>
</tr>
<tr>
<td>value</td>
<td>Value at the given position</td>
</tr>
</tbody>
</table>

NResult N_API NSfmMatrixGetColumnCount_D (HNSfmMatrixD phMatrix, int * columnCount)

NSfmMatrixGetColumnCount_D Returns column count in the matrix.

Returns:
NResult

Parameters:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>phMatrix</td>
<td>HNSfmMatrixD instance</td>
</tr>
<tr>
<td>columnCount</td>
<td>Contains number of columns available in the matrix</td>
</tr>
</tbody>
</table>

NResult N_API NSfmMatrixGetRowCount_D (HNSfmMatrixD phMatrix, int * rowCount)

NSfmMatrixGetRowCount_D Returns row count in the matrix.

Returns:
NResult

Parameters:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>phMatrix</td>
<td>HNSfmMatrixD instance</td>
</tr>
<tr>
<td>rowCount</td>
<td>Contains number of rows available in the matrix</td>
</tr>
</tbody>
</table>

**NSfmMesher.h File Reference**

**Macros**

#define SFM_NSFM_MESHER_H

**Functions**

NResult N_API NSfmMesherCreate (HNSfmMesher *phMesher)

Initializes HNSfmMesherImpl instance.

NResult N_API NSfmMesherDestroy (HNSfmMesher phMesher)

Destroys the HNSfmMesherImpl instance.

NResult N_API NSfmMesherReconstructFromFile (HNSfmMesher phMesher, SfmTaskResult *result, SfmMeshReconstructionFormat format, SfmMeshReconstructionResolution resolution, const char *pointCloudFile, HNSfmPointCloud pointCloud)

NSfmMesherReconstructFromFile Reconstructs the mesh using a set of points loaded from a file.
NSfmMesherReconstructFromPointCloud (HNSfmMesher phMesher, SfmTaskResult *result, SfmMeshReconstructionFormat format, SfmMeshReconstructionResolution resolution, HNSfmPointCloud pointCloud)
NSfmMesherReconstructFromPointCloud Reconstructs the mesh using a PointCloud.

Detailed Description
File containing mesh generation functions.

Author:
Neurotechnology

Date:
29 October 2014

Macro Definition Documentation
#define SFM_NSFM_MESHER_H

Function Documentation
NResult N_API NSfmMesherCreate (HNSfmMesher * phMesher)

Initializes HNSfmMesherImpl instance.

Parameters:
| phMesher | Pointer to be initialized. |

Returns:
Indicates success or failures

NResult N_API NSfmMesherDestroy (HNSfmMesher phMesher)

Destroys the HNSfmMesherImpl instance.

Parameters:
| phMesher | Pointer to the instance to be destroyed. |

Returns:
Indicates success or failures

NResult N_API NSfmMesherReconstructFromFile (HNSfmMesher phMesher, SfmTaskResult *result, SfmMeshReconstructionFormat format, SfmMeshReconstructionResolution resolution, const char *pointCloudFile, HNSfmPointCloud pointCloud)
NSfmMesherReconstructFromFile Reconstructs the mesh using a set of points loaded from a file.

Returns:
NResult Indicates success or failures
Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phMesher</td>
<td>HNSfmMesher instance</td>
</tr>
<tr>
<td>result</td>
<td>Reconstruction result</td>
</tr>
<tr>
<td>format</td>
<td>Mesh reconstruction format</td>
</tr>
<tr>
<td>resolution</td>
<td>Resolution to be used</td>
</tr>
<tr>
<td>pointCloudFile</td>
<td>Saved point cloud file to be used in reconstruction</td>
</tr>
<tr>
<td>pointCloud</td>
<td>PointCloud instance to fill resultant data</td>
</tr>
</tbody>
</table>

NResult N_API NSfmMesherReconstructFromPointCloud (HNSfmMesher phMesher, SfmTaskResult * result, SfmMeshReconstructionFormat format, SfmMeshReconstructionResolution resolution, HNSfmPointCloud pointCloud)

NSfmMesherReconstructFromPointCloud Reconstructs the mesh using a PointCloud.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phMesher</td>
<td>HNSfmMesher instance</td>
</tr>
<tr>
<td>result</td>
<td>Reconstruction result</td>
</tr>
<tr>
<td>format</td>
<td>Mesh reconstruction format</td>
</tr>
<tr>
<td>resolution</td>
<td>Resolution to be used</td>
</tr>
<tr>
<td>pointCloud</td>
<td>PointCloud instance to be used in reconstruction</td>
</tr>
</tbody>
</table>

NSfmPointCloud.h File Reference

Macros

#define SFM_NSFM_POINT_CLOUD_H

Functions

NResult N_API NSfmPointCloudCreate (HNSfmPointCloud *phCloud)
NSfmPointCloudCreate Initializes HNSfmPointCloudImpl instance.

NResult N_API NSfmPointCloudDestroy (HNSfmPointCloud phCloud)
NSfmPointCloudDestroy Destroys the HNSfmPointCloudImpl instance.

NResult N_API NSfmPointCloudImportDensePointsFromFile (HNSfmPointCloud phCloud, const char *fileName)
NSfmPointCloudImportDensePointsFromFile Loads dense pointcloud data form a file.

NResult N_API NSfmPointCloudSaveDensePointsToFile (HNSfmPointCloud phCloud, SfmPointCloudFormat format, const char *fileName)
NSfmPointCloudSaveDensePointsToFile Saves dense point cloud to a file.

NResult N_API NSfmPointCloudSaveSparsePointsToFile (HNSfmPointCloud phCloud, SfmPointCloudFormat format, const char *fileName)
NSfmPointCloudSaveSparsePointsToFile Saves dense point cloud to a file.

NResult N_API NSfmPointCloudSaveProjectionMatrices (HNSfmPointCloud phCloud, const char *rootFolder)
NSfmPointCloudSaveProjectionMatrices Save projection matrices used for point cloud generation to a file.

NResult N_API NSfmPointCloudSaveVisibilityData (HNSfmPointCloud phCloud, const char *rootFolder)
NSfmPointCloudSaveVisibilityData Save visibility information to file vis.dat

NResult N_API NSfmPointCloudCopyBaseImages (HNSfmPointCloud phCloud, const char *rootFolder)
NSfmPointCloudCopyBaseImages Copies base images used for point cloud generation to a specified folder.

NResult N_API NSfmPointCloudGenerateOutputForPMVS (HNSfmPointCloud phCloud, const char *rootFolder)
NSfmPointCloudGenerateOutputForPMVS Generates a file structure which can be used as an input for PMVS2 dense point cloud generator.

NResult N_API NSfmPointCloudSaveCameraPositions (HNSfmPointCloud phCloud, const char *fileName)
NSfmPointCloudSaveCameraPositions Save camera positions used for point cloud generation.

NResult N_API NSfmPointCloudGetCameraCount (HNSfmPointCloud phCloud, int *count)
GetCameraCount Returns total camera count used.

NResult N_API NSfmPointCloudGetCameraPositionMatrix (HNSfmPointCloud phCloud, HNSfmMatrixD *matrix, int cameraIndex)
NSfmPointCloudGetCameraPositionMatrix Returns camera position matrix for the given camera. matrix is NULL if camera is not being used by the algorithm.

NResult N_API NSfmPointCloudGetCameraRotationMatrix (HNSfmPointCloud phCloud, HNSfmMatrixD *matrix, int cameraIndex)
NSfmPointCloudGetCameraRotationMatrix Returns camera rotation matrix for the given camera. matrix is NULL if camera is not being used by the algorithm.

NResult N_API NSfmPointCloudGetCameraTranslationMatrix (HNSfmPointCloud phCloud, HNSfmMatrixD *matrix, int cameraIndex)
GetCameraTranslationMatrix Returns camera translation matrix for the given camera. matrix is NULL if camera is not being used by the algorithm.

NResult N_API NSfmPointCloudSaveBaseImageNames (HNSfmPointCloud phCloud, const char *fileName)
NSfmPointCloudSaveBaseImageNames Save base image names used for point cloud generation.

NResult N_API NSfmPointCloudGenerateOutputForMeshLab (HNSfmPointCloud phCloud, const char *cameraPositionsFilePath, const char *baseImageNamesFilePath)
NSfmPointCloudGenerateOutputForMeshLab Generates a file structure which can be used as an input for Meshlab texture mapping.

NResult N_API NSfmPointCloudSaveMeshToFile (HNSfmPointCloud phCloud, int meshIndex, SfmMeshExportFormat format, const char *fileName)
NSfmPointCloudSaveMeshToFile Save reconstructed mesh(es) to file(s)

NResult N_API NSfmPointCloudGetMeshCount (HNSfmPointCloud phCloud, int *count)
NSfmPointCloudGetMeshCount (HNSfmPointCloud phCloud, int *count)

Detailed Description
File containing point cloud information.

Author:
Neurotechnology

Date:
29 October 2014

Macro Definition Documentation
#define SFM_NSFM_POINT_CLOUD_H
Function Documentation

NResult N_API NSfmPointCloudCopyBaseImages (HNSfmPointCloud phCloud, const char * rootFolder)

NSfmPointCloudCopyBaseImages Copies base images used for point cloud generation to a specified folder.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phCloud</td>
<td>HNSfmPointCloudImpl instance</td>
</tr>
<tr>
<td>rootFolder</td>
<td>Output folder</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudCreate (HNSfmPointCloud * phCloud)

NSfmPointCloudCreate Initializes HNSfmPointCloudImpl instance.

Returns:
NResult Indicates success or failures

Parameters:
| phCloud     | Pointer to be initialized            |

NResult N_API NSfmPointCloudDestroy (HNSfmPointCloud phCloud)

NSfmPointCloudDestroy Destroys the HNSfmPointCloudImpl instance.

Returns:
NResult Indicates success or failures

Parameters:
| phCloud     | Pointer to the instance to be destroyed. |

NResult N_API NSfmPointCloudGenerateOutputForMeshLab (HNSfmPointCloud phCloud, const char * cameraPositionsFilePath, const char * baseImageNamesFilePath)

NSfmPointCloudGenerateOutputForMeshLab Generates a file structure which can be used as an input for Meshlab texture mapping.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phCloud</th>
<th>HNSfmPointCloudImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>cameraPositionsFilePath</td>
<td>File name to write camera positions</td>
</tr>
<tr>
<td>baseImageNamesFilePath</td>
<td>File name to write base image names</td>
</tr>
</tbody>
</table>
NResult N_API NSfmPointCloudGenerateOutputForPMVS (HNSfmPointCloud phCloud, const char * rootFolder)

NSfmPointCloudGenerateOutputForPMVS Generates a file structure which can be used as an input for PMVS2 dense point cloud generator.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phCloud</th>
<th>HNSfmPointCloudImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>rootFolder</td>
<td>Output folder</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudGetCameraCount (HNSfmPointCloud phCloud, int * count)

GetCameraCount Returns total camera count used.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phCloud</th>
<th>HNSfmPointCloudImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Camera count</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudGetCameraPositionMatrix (HNSfmPointCloud phCloud, HNSfmMatrixD * matrix, int cameraIndex)

NSfmPointCloudGetCameraPositionMatrix Returns camera position matrix for the given camera. matrix is NULL if camera is not being used by the algorithm.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phCloud</th>
<th>HNSfmPointCloudImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>matrix</td>
<td>position matrix</td>
</tr>
<tr>
<td>cameraIndex</td>
<td>Index of the camera</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudGetCameraRotationMatrix (HNSfmPointCloud phCloud, HNSfmMatrixD * matrix, int cameraIndex)

NSfmPointCloudGetCameraRotationMatrix Returns camera rotation matrix for the given camera. matrix is NULL if camera is not being used by the algorithm.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phCloud</th>
<th>HNSfmPointCloudImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>matrix</td>
<td>rotation matrix</td>
</tr>
<tr>
<td>cameraIndex</td>
<td>Index of the camera</td>
</tr>
</tbody>
</table>
NResult N_API NSfmPointCloudGetCameraTranslationMatrix (HNSfmPointCloud phCloud, HNSfmMatrixD * matrix, int cameraIndex)

GetCameraTranslationMatrix Returns camera translation matrix for the given camera. matrix is NULL if camera is not being used by the algorithm.

**Returns:**
NResult Indicates success or failures

**Parameters:**
<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phCloud</td>
<td>HNSfmPointCloudImpl instance</td>
</tr>
<tr>
<td>matrix</td>
<td>translation matrix</td>
</tr>
<tr>
<td>cameraIndex</td>
<td>Index of the camera</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudGetMeshCount (HNSfmPointCloud phCloud, int * count)

NSfmPointCloudSaveMeshToFile Save reconstructed mesh(es) to file(s)

**Returns:**
NResult Indicates success or failures

**Parameters:**
<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phCloud</td>
<td>HNSfmPointCloudImpl instance</td>
</tr>
<tr>
<td>count</td>
<td>Mesh count</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudImportDensePointsFromFile (HNSfmPointCloud phCloud, const char * fileName)

NSfmPointCloudImportDensePointsFromFile Loads dense pointcloud data from a file.

**Returns:**
NResult

**Parameters:**
<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phCloud</td>
<td>HNSfmPointCloudImpl instance</td>
</tr>
<tr>
<td>fileName</td>
<td>Filename to load points</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudSaveBaseImageNames (HNSfmPointCloud phCloud, const char * fileName)

NSfmPointCloudSaveBaseImageNames Save base image names used for point cloud generation.

**Returns:**
NResult Indicates success or failures

**Parameters:**
<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phCloud</td>
<td>HNSfmPointCloudImpl instance</td>
</tr>
<tr>
<td>fileName</td>
<td>Output file name</td>
</tr>
</tbody>
</table>
NResult N_API NSfmPointCloudSaveCameraPositions (HNSfmPointCloud *phCloud, const char *fileName)

NSfmPointCloudSaveCameraPositions Save camera positions used for point cloud generation.

Returns:
NResult Indicates success or failures

Parameters:
| phCloud | HNSfmPointCloudImpl instance |
| fileName | Output file name |

NResult N_API NSfmPointCloudSaveDensePointsToFile (HNSfmPointCloud *phCloud, SfmPointCloudFormat format, const char *fileName)

NSfmPointCloudSaveDensePointsToFile Saves dense point cloud to a file.
NSfmPointCloudSaveToFile Saves dense point cloud to a file.

Returns:
NResult

Parameters:
| phCloud | HNSfmPointCloudImpl instance |
| format | File format to be used |
| fileName | Save file name |

NResult N_API NSfmPointCloudSaveMeshToFile (HNSfmPointCloud *phCloud, int meshIndex, SfmMeshExportFormat format, const char *fileName)

NSfmPointCloudSaveMeshToFile Save reconstructed mesh(es) to file(s)

Returns:
NResult Indicates success or failures

Parameters:
| phCloud | HNSfmPointCloudImpl instance |
| meshIndex | Index of the mesh to be saved |
| format | Output format |
| fileName | Output file name |

NResult N_API NSfmPointCloudSaveProjectionMatrices (HNSfmPointCloud *phCloud, const char *rootFolder)

NSfmPointCloudSaveProjectionMatrices Save projection matrices used for point cloud generation to a file.

Returns:
NResult Indicates success or failures

Parameters:
| phCloud | HNSfmPointCloudImpl instance |
| rootFolder | Output folder |
NResult N_API NSfmPointCloudSaveVisibilityData (HNSfmPointCloud phCloud, const char * rootFolder)

NSfmPointCloudSaveVisibilityData Save visibility information to file vis.dat

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>phCloud</th>
<th>HNSfmPointCloudImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>rootFolder</td>
<td>Output folder</td>
</tr>
</tbody>
</table>

NResult N_API NSfmPointCloudSaveSparsePointsToFile (HNSfmPointCloud phCloud, SfmPointCloudFormat format, const char * fileName)

NSfmPointCloudSaveSparsePointsToFile Saves dense point cloud to a file.

Returns:
NResult

Parameters:

<table>
<thead>
<tr>
<th>phCloud</th>
<th>HNSfmPointCloudImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>File format to be used</td>
</tr>
<tr>
<td>fileName</td>
<td>Save file name</td>
</tr>
</tbody>
</table>

NSfmTaskProgress.h File Reference

Macros
#define SFM_NSFM_TASK_PROGRESS_H

Functions
NResult N_API NSfmTaskProgressCreate (HNSfmTaskProgress *phProgress)

NSfmTaskProgressCreate Initializes HNSfmTaskProgressImpl instance.

NResult N_API NSfmTaskProgressDestroy (HNSfmTaskProgress phProgress)

NSfmTaskProgressDestroy Destroys the HNSfmTaskProgress instance.

NResult N_API NSfmTaskProgressGetCurrent (HNSfmTaskProgress phProgress, SfmTask *currentTask, int *progress)

NSfmTaskProgressGetCurrent Returns current task and its progress.

Detailed Description
File containing async task progress.

Author:
Neurotechnology
Macro Definition Documentation

#define SFM_NSFM_TASK_PROGRESS_H

Function Documentation

NResult N_API NSfmTaskProgressCreate (HNSfmTaskProgress * phProgress)

NSfmTaskProgressCreate Initializes HNSfmTaskProgressImpl instance.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phProgress</td>
<td>Pointer to be initialized</td>
</tr>
</tbody>
</table>

NResult N_API NSfmTaskProgressDestroy (HNSfmTaskProgress phProgress)

NSfmTaskProgressDestroy Destroys the HNSfmTaskProgress instance.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phProgress</td>
<td>Pointer to the instance to be destroyed.</td>
</tr>
</tbody>
</table>

NResult N_API NSfmTaskProgressGetCurrent (HNSfmTaskProgress phProgress, SfmTask * currentTask, int * progress)

NSfmTaskProgressGetCurrent Returns current task and its progress.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phProgress</td>
<td>HNSfmTaskProgress instance</td>
</tr>
<tr>
<td>SfmTask*</td>
<td>current task</td>
</tr>
<tr>
<td>int*</td>
<td>progress of current task</td>
</tr>
</tbody>
</table>

NSfmTypes.h File Reference

Macros

#define NSFM_TYPES_H

Typedefs
typedef enum SfmPointCloudFormat_ SfmPointCloudFormat
    Supported Export formats for dense point cloud.
typedef enum SfmMeshReconstructionFormat_ SfmMeshReconstructionFormat
    Supported mesh reconstruction algorithms.
typedef enum SfmMeshReconstructionResolution_ SfmMeshReconstructionResolution
    Supported mesh reconstruction resolutions.
typedef enum SfmMeshExportFormat_ SfmMeshExportFormat
    Supported mesh export formats.
typedef enum SfmTaskResult_ SfmTaskResult
    Possible return values for functions.
typedef enum SfmTask_ SfmTask
    Represents a particular task performed by the point cloud generation algorithm.
typedef void * HNPoints
typedef void * HNModelParameters
typedef void * HNSfmPointCloud
typedef void * HNSfmMesher
typedef void * HNSfmTexturer
typedef void * HNStructureFromMotion
typedef void * HNSfmTaskProgress
typedef void * HNSfmDensePointGenerationOptions
typedef void * HNSfmMatrix
typedef void * HNSfmMatrixD
typedef void (* OnAsyncGenerationEndCallback )(SfmTaskResult)

Enumerations
enum SfmPointCloudFormat_ { ASC, PLY }
    Supported Export formats for dense point cloud. enum SfmMeshReconstructionFormat_ { Poisson }
    Supported mesh reconstruction algorithms. enum SfmMeshReconstructionResolution_ { Low = 0, High }
    Supported mesh reconstruction resolutions. enum SfmMeshExportFormat_ { OBJ }
    Supported mesh export formats. enum SfmTaskResult_ { Error = 1, NoImagesToProcess = 2, TooFewMatches = 4, NoSuitableImagePairs = 8, OutOfMemory = 16, IOError = 32, InvalidImageFile = 64, NoLicense = 128, NoPermission = 256, MeshSimplificationFailed = 512, MeshReconstructionFailed = 1024, Success = 2048 }
    Possible return values for functions. enum SfmTask_ { None = 1, FeatureExtraction = 2, FeatureMatching = 4, PopulatingTracks = 8, FilteringTracks = 16, MatchingFeatures = 32, Two_ViewReconstruction = 64, AddingMoreFrames = 128, GeneratingOutput = 256 }
    Represents a particular task performed by the point cloud generation algorithm.

Macro Definition Documentation
#define NSFM_TYPES_H

Typedef Documentation
typedef enum SfmMeshExportFormat_ SfmMeshExportFormat
    Supported mesh export formats.
typedef enum SfmMeshReconstructionFormat_ SfmMeshReconstructionFormat
    Supported mesh reconstruction algorithms.
typedef enum SfmMeshReconstructionResolution_ SfmMeshReconstructionResolution

Supported mesh reconstruction resolutions.

typedef enum SfmPointCloudFormat_ SfmPointCloudFormat

Supported Export formats for dense point cloud.

typedef enum SfmTask_ SfmTask

Represents a particular task performed by the point cloud generation algorithm.

typedef enum SfmTaskResult_ SfmTaskResult

Possible return values for functions.

---

**Enumeration Type Documentation**

**enum SfmMeshExportFormat_**

Supported mesh export formats.

**Enumerator**

OBJ

**enum SfmMeshReconstructionFormat_**

Supported mesh reconstruction algorithms.

**Enumerator**

Poisson

**enum SfmMeshReconstructionResolution_**

Supported mesh reconstruction resolutions.

**Enumerator**

Low
High Low resolution

**enum SfmPointCloudFormat_**

Supported Export formats for dense point cloud.

**Enumerator**

ASC
PLY ASC file format

**enum SfmTask_**

Represents a particular task performed by the point cloud generation algorithm.
 Enumerator

 None
 FeatureExtraction
 FeatureMatching
 PopulatingTracks
 FilteringTracks
 MatchingFeatures
 Two_ViewReconstruction
 AddingMoreFrames
 GeneratingOutput

 enum SfmTaskResult_

 Possible return values for functions.

 Enumerator

 Error
 NoImagesToProcess Unspecified error
 TooFewMatches There are no images which can be used for point cloud generation
 NoSuitableImagePairs Point cloud generation failed due to few matches
 OutOfMemory Point cloud generation failed due to no suitable image pairs
 IOError Out of memory
 InvalidImageFile IO error
 NoLicense Image file cannot be loaded or file contains invalid images
 NoPermission No License
 MeshSimplificationFailed No Permission
 MeshReconstructionFailed Mesh simplification failed
 Success Mesh Reconstruction failed

 NStructureFromMotion.h File Reference

 Macros
 #define SFM_INTERNAL_STRUCTURE_FROM_MOTION_H

 Functions
 NResult N_API NSfmStructureFromMotionCreate (HNStructureFromMotion *phSfm)
 Initializes HNStructureFromMotionImpl instance.
 NResult N_API NSfmStructureFromMotionDestroy (HNStructureFromMotion phSfm)
 Destroy the HNStructureFromMotionImpl instance.
 NResult N_API NSfmSetDensePointGenerationOptions (HNStructureFromMotion hSfm, const
 HNSfmDensePointGenerationOptions options)
 NSfmSetDensePointGenerationOptions Set dense point cloud generation options.
 NResult N_API NSfmLoadDensePointGenerationOptions (HNStructureFromMotion hSfm, const char
 *optionsFileName)
 NSfmSetDensePointGenerationOptions Set dense point cloud generation options.
 NResult N_API NSfmGetDensePointGenerationOptions (HNStructureFromMotion hSfm, const
 HNSfmDensePointGenerationOptions *options)
 NSfmGetDensePointGenerationOptions Get dense point cloud generation options.
NResult N_API NSfmGetPointCloud (HNStructureFromMotion hSfm, HNSfmPointCloud *pointCloud)

NSfmAddImage Adds a candidate image for point cloud generation.

NResult N_API NSfmSetTrackLength (HNStructureFromMotion hSfm, int trackLength)

NSfmSetTrackLength sets the track length.

NResult N_API NSfmGetTrackLength (HNStructureFromMotion hSfm, int *trackLength)

NSfmGetTrackLength returns the track length.

NResult N_API NSfmSetMinTriangularThreshold (HNStructureFromMotion hSfm, int minTriangularThreshold)

NSfmSetMinTriangularThreshold sets the minimum triangulation threshold.

NResult N_API NSfmGetMinTriangularThreshold (HNStructureFromMotion hSfm, int *minTriangularThreshold)

NSfmGetMinTriangularThreshold returns the minimum triangulation threshold.

NResult N_API NSfmSetSequenceLength (HNStructureFromMotion hSfm, int sequenceLength)

NSfmSetSequenceLength sets the sequence length. If Sequence is on, image set will be divided into mini-batches by size of sequence length.

NResult N_API NSfmGetSequenceLength (HNStructureFromMotion hSfm, int *sequenceLength)

NSfmGetSequenceLength returns the sequence length.

NResult N_API NSfmSetSequenceOnOff (HNStructureFromMotion hSfm, NBool isOn)

NSfmSetSequenceOnOff sets the sequence on or off. If off, sequence length will be ignored and images will be compared by pairs.

NResult N_API NSfmGetSequenceOnOff (HNStructureFromMotion hSfm, NBool *isOn)

NSfmGetSequenceOnOff returns current value for SequenceOnOff.

NResult N_API NSfmGeneratePointCloudFromFolder_Async (HNStructureFromMotion hSfm, OnAsyncGenrationEndCallback callback, const char *hFolderName)

NSfmGeneratePointCloudFromFolder_Async Generate point cloud from contents of an image folder asynchronously.

NResult N_API NSfmGeneratePointCloudFromFile_Async (HNStructureFromMotion hSfm, OnAsyncGenrationEndCallback callback, const char *hFileName)

NSfmGeneratePointCloudFromFile_Async Generate point cloud from contents of a text file containing image names asynchronously.

NResult N_API NSfmGetAsyncTaskProgress (HNStructureFromMotion hSfm, HNSfmTaskProgress *progress)

NSfmGetAsyncTaskProgress returns progress of the asynchronous task.

NResult N_API NSfmGeneratePointCloudFromFolder_Sync (HNStructureFromMotion hSfm, SfmTaskResult *result, const char *hFolderName)

NSfmGeneratePointCloudFromFolder_Sync Generate point cloud from contents of an image folder synchronously.

NResult N_API NSfmGeneratePointCloudFromFile_Sync (HNStructureFromMotion hSfm, SfmTaskResult *result, const char *hFileName)

NSfmGeneratePointCloudFromFile_Sync Generate point cloud from contents of a text file containing image names synchronously.

---

Macro Definition Documentation

#define SFM_INTERNAL_STRUCTURE_FROM_MOTION_H

Function Documentation
NResult N_API NSfmGeneratePointCloudFromFile_Async (HNStructureFromMotion hSfm, OnAsyncGenerationEndCallback callback, const char * hFileName)

NSfmGeneratePointCloudFromFile_Async Generate point cloud from contents of an text file containing image names asynchronously.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phSfm</th>
<th>HNStructureFromMotionImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>callback</td>
<td>Callback will be executed after the point cloud generation finished</td>
</tr>
<tr>
<td>hFileName</td>
<td>File name containing image names</td>
</tr>
</tbody>
</table>

NResult N_API NSfmGeneratePointCloudFromFile_Sync (HNStructureFromMotion hSfm, SfmTaskResult * result, const char * hFileName)

NSfmGeneratePointCloudFromFile_Sync Generate point cloud from contents of an text file containing image names synchronously.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phSfm</th>
<th>HNStructureFromMotionImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>result</td>
<td>Point cloud generation result</td>
</tr>
<tr>
<td>hFileName</td>
<td>File name containing image names</td>
</tr>
</tbody>
</table>

NResult N_API NSfmGeneratePointCloudFromFolder_Async (HNStructureFromMotion hSfm, OnAsyncGenerationEndCallback callback, const char * hFolderName)

NSfmGeneratePointCloudFromFolder_Async Generate point cloud from contents of an image folder asynchronously.

Returns:
NResult Indicates success or failures

Parameters:
<table>
<thead>
<tr>
<th>phSfm</th>
<th>HNStructureFromMotionImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>callback</td>
<td>Callback will be executed after the point cloud generation finished</td>
</tr>
<tr>
<td>hFolderName</td>
<td>Image folder name</td>
</tr>
</tbody>
</table>

NResult N_API NSfmGeneratePointCloudFromFolder_Sync (HNStructureFromMotion hSfm, SfmTaskResult * result, const char * hFolderName)

NSfmGeneratePointCloudFromFolder_Sync Generate point cloud from contents of an image folder synchronously.

Returns:
NResult Indicates success or failures
Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phSfm</td>
<td>HNStructureFromMotionImpl instance</td>
</tr>
<tr>
<td>result</td>
<td>Point cloud generation result</td>
</tr>
<tr>
<td>hFolderName</td>
<td>Image folder name</td>
</tr>
</tbody>
</table>

NResult N_API NSfmGetAsyncTaskProgress (HNStructureFromMotion hSfm, HNSfmTaskProgress * progress)

NSfmGetAsyncTaskProgress Returns progress of the asynchronous task.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phSfm</td>
<td>HNStructureFromMotionImpl instance</td>
</tr>
<tr>
<td>progress</td>
<td>Task progress</td>
</tr>
</tbody>
</table>

NResult N_API NSfmGetDensePointGenerationOptions (HNStructureFromMotion hSfm, HNSfmDensePointGenerationOptions * options)

NSfmGetDensePointGenerationOptions Get dense point cloud generation options.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>phSfm</td>
<td>HNStructureFromMotionImpl instance</td>
</tr>
<tr>
<td>options</td>
<td>Option values will be filled</td>
</tr>
</tbody>
</table>

NResult N_API NSfmGetMinTriangularThreshold (HNStructureFromMotion hSfm, int * minTriangularThreshold)

NSfmGetMinTriangularThreshold returns the minimum triangulation threshold.

Returns:
NResult Indicates success or failures

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>minTriangularThreshold</td>
<td>current minimum triangulation threshold value used in the algorithm</td>
</tr>
</tbody>
</table>

NResult N_API NSfmGetPointCloud (HNStructureFromMotion hSfm, HNSfmPointCloud * pointCloud)

NSfmAddImage Adds a candidate image for point cloud generation.

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image</td>
<td>NImage NSfmGetPointCloud returns the point cloud generated using candidate images</td>
</tr>
</tbody>
</table>
Returns:
NResult Indicates success or failures

Parameters:

| *pointCloude* | point cloud generated using candidate images |

NResult N_API NSfmGetSequenceLength (HNStructureFromMotion *hSfm, int *sequenceLength)

NSfmGetSequenceLength Returns the sequence length.

Returns:
NResult Indicates success or failures

Parameters:

| *sequenceLength* | Current sequence length value used in the algorithm |

NResult N_API NSfmGetSequenceOnOff (HNStructureFromMotion *hSfm, NBool *isOn)

NSfmGetSequenceOnOff Returns current value for SequenceOnOff.

Returns:
NResult Indicates success or failures

Parameters:

| isOn | Sequence length is ignored or not |

NResult N_API NSfmGetTrackLength (HNStructureFromMotion *hSfm, int *trackLength)

NSfmGetTrackLength returns the track length.

Returns:
NResult Indicates success or failures

Parameters:

| trackLength | current track length value used in the algorithm |

NResult N_API NSfmLoadDensePointGenerationOptions (HNStructureFromMotion *hSfm, const char *optionsFileName)

NSfmSetDensePointGenerationOptions Set dense point cloud generation options.

Returns:
NResult Indicates success or failures

Parameters:

| *phSfm* | HNStructureFromMotionImpl instance |
| optionsFileName | Text file containing dense point cloud generation options |

NResult N_API NSfmSetDensePointGenerationOptions (HNStructureFromMotion *hSfm, const HNSfmDensePointGenerationOptions *options)
NSfmSetDensePointGenerationOptions Set dense point cloud generation options.

**Retours:**
NResult Indicates success or failures

**Parameters:**

<table>
<thead>
<tr>
<th>phSfm</th>
<th>HNStructureFromMotionImpl instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>options</td>
<td>Option values to be set</td>
</tr>
</tbody>
</table>

NResult N_API NSfmSetMinTriangularThreshold (HNStructureFromMotion hSfm, int minTriangularThreshold)

NSfmSetMinTriangularThreshold set the minimum triangulation threshold.

**Retours:**
NResult Indicates success or failures

**Parameters:**

| minTriangularThreshold | minimum triangulation threshold to be used in the algorithm |

NResult N_API NSfmSetSequenceLength (HNStructureFromMotion hSfm, int sequenceLength)

NSfmSetSequenceLength Set the sequence length. If Sequence is on, image set will be divided in to mini batches by size of sequence length.

**Retours:**
NResult Indicates success or failures

**Parameters:**

| sequenceLength | Sequence length to be used in the algorithm |

NResult N_API NSfmSetSequenceOnOff (HNStructureFromMotion hSfm, NBool isOn)

NSfmSetSequenceOnOff Set the sequence on or off. If off, sequence length will be ignored and images will be compared by pairs.

**Retours:**
NResult Indicates success or failures

**Parameters:**

| isOn | Sequence length is ignored or not |

NResult N_API NSfmSetTrackLength (HNStructureFromMotion hSfm, int trackLength)

NSfmSetTrackLength set the track length.

**Retours:**
NResult Indicates success or failures
Parameters:

| trackLength | track length to be used in the algorithm |

NResult N_API NSfmStructureFromMotionCreate (HNStructureFromMotion * phSfm)

Initializes HNStructureFromMotionImpl instance.

Returns:
Indicates success or failures

Parameters:

| phSfm | Pointer to be initialized |

NResult N_API NSfmStructureFromMotionDestroy (HNStructureFromMotion phSfm)

Destroy the HNStructureFromMotionImpl instance.

Returns:
NResult Indicates success or failures

Parameters:

| phSfm | Pointer to the instance to be destroyed |

.Net Reference

Namespace List
Here is a list of all namespaces with brief descriptions:

- Neurotec
- Neurotec.SentiSculpt
- Neurotec.SentiSculpt.Exceptions
- Neurotec.SentiSculpt.Base

Class List
Here are the classes, structs, unions and interfaces with brief descriptions:

- Neurotec.SentiSculpt.NSfmDensePointGenerationOptions
- Neurotec.SentiSculpt.Exceptions.NSfmException (Used for exception handling)
- Neurotec.SentiSculpt.NSfmMatrix<T>
- Neurotec.SentiSculpt.NSfmMesher (Used for mesh reconstruction)
- Neurotec.SentiSculpt.NSfmObject
- Neurotec.SentiSculpt.NSfmPointCloud (Point cloud implementation)
- Neurotec.SentiSculpt.NSfmTaskProgress
- NSfmTexturer (Point cloud implementation)
- Neurotec.SentiSculpt.NStructureFromMotion (Point cloud implementation)
Neurotec.SentiSculpt.NSfmDensePointGenerationOptions Class Reference

Inherits Neurotec.SentiSculpt.NSfmObject.

Public Member Functions

NSfmDensePointGenerationOptions ()
NSfmDensePointGenerationOptions (HNSfmDensePointGenerationOptions safeHandle)

int GetLevel ()
    GetLevel Returns option level.

void SetLevel (int level)
    SetLevel Sets option level.

int GetCsize ()
    GetLevel Returns option csize.

void SetCsize (int value)
    SetLevel Sets option csize.

float GetThreshold ()
    GetLevel Returns option threshold.

void SetThreshold (float value)
    SetLevel Sets option threshold.

int GetWsize ()
    GetLevel Returns option wsize.

void SetWsize (int value)
    SetLevel Sets option wsize.

int GetMinImageNum ()
    GetLevel Returns option minImageNum.

void SetMinImageNum (int value)
    SetLevel Sets option minImageNum.

int GetCpu ()
    GetLevel Returns option cpu.

void SetCpu (int value)
    SetLevel Sets option cpu.

float GetQuad ()
    GetQuad Returns option quad.

void SetQuad (float value)
    SetQuad Sets option quad.

int GetMaxAngle ()
    GetMaxAngle Returns option max angle.

void SetMaxAngle (int value)
    SetMaxAngle Sets option angle.

int GetUseVisData ()
    GetLevel Returns option useVisData.

void SetUseVisData (int value)
    SetLevel Sets option useVisData.

int GetSequence ()
    GetLevel Returns option sequence.
void SetSequence (int value)

SetLevel Sets option sequence.

Additional Inherited Members

Constructor & Destructor Documentation

Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.NSfmDensePointGenerationOptions ()

Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.NSfmDensePointGenerationOptions (HNSfmDensePointGenerationOptions safeHandle)

Member Function Documentation

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetCpu ()

GetLevel Returns option cpu.

Returns:
    int cpu

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetCsize ()

GetLevel Returns option csize.

Returns:
    int csize

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetLevel ()

GetLevel Returns option level.

Returns:
    int Level

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetMinImageNum ()

GetLevel Returns option minImageNum.

Returns:
    int minImageNum

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetSequence ()

GetLevel Returns option sequence.

Returns:
    int sequence
float Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetQuad ()

    GetQuad Returns option quad.

    Returns:
    float quad

float Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetThreshold ()

    GetLevel Returns option threshold.

    Returns:
    float threshold

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetMaxAngle ()

    GetmaxAngle Returns option max angle.

    Returns:
    int max angle

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetUseVisData ()

    GetLevel Returns option useVisData.

    Returns:
    int useVisData

int Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.GetWsize ()

    GetLevel Returns option wsize.

    Returns:
    int wsize

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetCpu (int value)

    SetLevel Sets option cpu.

    Parameters:

    | int | value to be assigned for option cpu |

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetCsize (int value)

    SetLevel Sets option csize.
Parameters:

| int | value to be assigned for option csize |

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetLevel (int level)

SetLevel Sets option level.

Parameters:

| int | value to be assigned for option level |

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetMinImageNum (int value)

SetLevel Sets option minImageNum.

Parameters:

| int | value to be assigned for option minImageNum |

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetSequence (int value)

SetLevel Sets option sequence.

Parameters:

| int | value to be assigned for option sequence |

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetQuad (float value)

SetQuad Sets option quad.

Parameters:

| float | value to be assigned for option quad |

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetThreshold (float value)

SetLevel Sets option threshold.

Parameters:

| float | value to be assigned for option threshold |

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetMaxAngle (int value)

setMaxAngle Sets option max angle.

Parameters:

| int | value to be assigned for option max angle |
void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetUseVisData (int value)

SetLevel Sets option useVisData.

**Parameters:**

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option useVisData</td>
</tr>
</tbody>
</table>

void Neurotec.SentiSculpt.NSfmDensePointGenerationOptions.SetWsize (int value)

SetLevel Sets option wsize.

**Parameters:**

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>int</td>
<td>value to be assigned for option wsize</td>
</tr>
</tbody>
</table>

---

**Neurotec.SentiSculpt.Exceptions.NSfmException Class Reference**

Used for exception handling.

Inherits Exception.

**Public Member Functions**

NSfmException (SfmTaskResult result)
NSfmException (SfmTaskResult result, string message)
NSfmException (SfmTaskResult result, string message, Exception inner)

**Properties**

SfmTaskResult Code [get, set]

---

**Detailed Description**

Used for exception handling.

**Author:**

Neurotechnology

**Date:**

18 December 2014

---

**Constructor & Destructor Documentation**

Neurotec.SentiSculpt.Exceptions.NSfmException.NSfmException (SfmTaskResult result)

Neurotec.SentiSculpt.Exceptions.NSfmException.NSfmException (SfmTaskResult result, string message)

Neurotec.SentiSculpt.Exceptions.NSfmException.NSfmException (SfmTaskResult result, string message, Exception inner)

---

**Property Documentation**

47
The documentation for this class was generated from the following file:
NSfmException.cs

**Neurotec.SentiSculpt.NSfmMatrix\(<\ T\ >\)** Class Template Reference

Inherits **Neurotec.SentiSculpt.NSfmObject**.

**Public Member Functions**
- NSfmMatrix()
- NSfmMatrix(HNSfmMatrix safeHandle)
- int GetRowCount()
- int GetColumnCount()
- T Get(int rowId, int colId)

**Additional Inherited Members**

**Constructor & Destructor Documentation**

- Neurotec.SentiSculpt.NSfmMatrix\(<\ T\ >\).NSfmMatrix()
- Neurotec.SentiSculpt.NSfmMatrix\(<\ T\ >\).NSfmMatrix(HNSfmMatrix\(<\ T\ >\) safeHandle)

**Member Function Documentation**

- T Neurotec.SentiSculpt.NSfmMatrix\(<\ T\ >\).Get(int rowId, int colId)
- int Neurotec.SentiSculpt.NSfmMatrix\(<\ T\ >\).GetColumnCount()
- int Neurotec.SentiSculpt.NSfmMatrix\(<\ T\ >\).GetRowCount()

**Neurotec.SentiSculpt.NSfmMesher Class Reference**

Used for mesh reconstruction.
Inherits **Neurotec.SentiSculpt.NSfmObject**.

**Public Member Functions**
- NSfmMesher()
  
  **NSfmMesher** Constructor.
- NSfmMesher(HNSfmMesher safeHandle)
  
  **NSfmMesher** Constructor.

SfmTaskResult ReconstructFromFile (NSfmMeshReconstructionFormat format, NSfmMeshReconstructionResolution resolution, String pointCloudFile, NSfmPointCloud pointCloud)

ReconstructFromFile Reconstructs the mesh using a set of points loaded from a file.
SfmTaskResult ReconstructFromPointCloud (NSfmMeshReconstructionFormat format, NSfmMeshReconstructionResolution resolution, NSfmPointCloud pointCloud)

ReconstructFromPointCloud Reconstructs the mesh using a PointCloud.

Additional Inherited Members

Detailed Description
Used for mesh reconstruction.

Author:
Neurotechnology

Date:
18 December 2014

Constructor & Destructor Documentation

Neurotec.SentiSculpt.NSfmMesher.NSfmMesher ()

NSfmMesher Constructor.

Neurotec.SentiSculpt.NSfmMesher.NSfmMesher (HNSfmMesher safeHandle)

NSfmMesher Constructor.

Parameters:
safeHandle Underlying mesher implementation

Member Function Documentation

SfmTaskResult Neurotec.SentiSculpt.NSfmMesher.ReconstructFromFile
(NSfmMeshReconstructionFormat format, NSfmMeshReconstructionResolution resolution, String pointCloudFile, NSfmPointCloud pointCloud)

ReconstructFromFile Reconstructs the mesh using a set of points loaded from a file.

Returns:
SfmTaskResult Reconstruction result

Parameters:

<table>
<thead>
<tr>
<th>format</th>
<th>Mesh reconstruction format</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolution</td>
<td>Resolution to be used</td>
</tr>
<tr>
<td>pointCloudFile</td>
<td>Reconstructed mesh is saved in to pointCloudFile</td>
</tr>
<tr>
<td>pointCloud</td>
<td>PointCloud instance to fill resultant data</td>
</tr>
</tbody>
</table>

SfmTaskResult Neurotec.SentiSculpt.NSfmMesher.ReconstructFromPointCloud
(NSfmMeshReconstructionFormat format, NSfmMeshReconstructionResolution resolution, NSfmPointCloud pointCloud)

ReconstructFromPointCloud Reconstructs the mesh using a PointCloud.
Returns:
SfmTaskResult Reconstruction result

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Mesh reconstruction format</td>
</tr>
<tr>
<td>resolution</td>
<td>Resolution to be used</td>
</tr>
<tr>
<td>pointCloud</td>
<td>PointCloud instance to be used in reconstruction</td>
</tr>
</tbody>
</table>

Neurotec.SentiSculpt.NSfmObject Class Reference

Inherits IDisposable.

Public Member Functions

void Dispose()

Properties

HNSfmObject SafeHandle [get]

Member Function Documentation

void Neurotec.SentiSculpt.NSfmObject.Dispose ()

Property Documentation

HNSfmObject Neurotec.SentiSculpt.NSfmObject.SafeHandle [get]

Neurotec.SentiSculpt.NSfmPointCloud Class Reference

Point cloud implementation.
Inherits Neurotec.SentiSculpt.NSfmObject.

Public Member Functions

NSfmPointCloud ()
NSfmPointCloud (HNSfmPointCloud safeHandle)
void ImportDensePointsFromFile (String fileName)

ImportDensePointsFromFile Loads dense pointcloud data form a file.

void SaveDenseCloudToEngle (NSfmPointCloudFormat format, String file)

SaveDenseCloudToEngle Saves dense point cloud to a file.

void SaveSparseCloudToFile (NSfmPointCloudFormat format, String file)

SaveSparseCloudToFile Saves sparse point cloud to a file.

void SaveProjectionMatrices (String rootFolder)
SaveProjectionMatrices Save projection matrices used for point cloud generation to a file.

void SaveVisibilityData (String rootFolder)
SaveVisibilityData Save visibility information to file vis.dat.

void CopyBaseImages (String rootFolder)
CopyBaseImages Copies base images used for point cloud generation to a specified folder.

void GenerateOutputForPMVS (String rootFolder)
GenerateOutputForPMVS Generates a file structure which can be used as an input for PMVS2 dense point cloud generator.

void SaveCameraPositions (String fileName)
SaveCameraPositions Save camera positions used for point cloud generation.

int GetCameraCount ()
GetCameraCount Returns total camera count used.

NSfmMatrix< double > GetCameraPositionMatrix (int index)
GetCameraPositionMatrix Returns camera position matrix for the given camera. Returns NULL is camera is not being used by the algorithm.

NSfmMatrix< double > GetCameraRotationMatrix (int index)
GetCameraRotationMatrix Returns camera rotation matrix for the given camera. Returns NULL is camera is not being used by the algorithm.

NSfmMatrix< double > CameraTranslationMatrix (int index)
GetCameraTranslationMatrix Returns camera translation matrix for the given camera. Returns NULL is camera is not being used by the algorithm.

void SaveBaseImageNames (String fileName)
SaveBaseImageNames Save base image names used for point cloud generation.

void GenerateOutputForMeshLab (String cameraPositionsFilePath, String baseImageNamesFilePath)
GenerateOutputForMeshLab Generates a file structure which can be used as an input for Meshlab texture mapping.

void SaveMeshToFile (int meshIndex, NSfmMeshExportFormat format, String fileName)
SaveMeshToFile Save reconstructed mesh(es) to file(s)

int GetMeshCount ()
GetMeshCount Returns mesh count.

Additional Inherited Members

Detailed Description
Point cloud implementation.

Author:
Neurotechnology

Date:
18 December 2014

Constructor & Destructor Documentation

Neurotec.SentiSculpt.NSfmPointCloud.NSfmPointCloud ()

Neurotec.SentiSculpt.NSfmPointCloud.NSfmPointCloud (HNSfmPointCloud safeHandle)

Member Function Documentation
NSfmMatrix<double> Neurotec.SentiSculpt.NSfmPointCloud.CameraTranslationMatrix (int index)

GetCameraTranslationMatrix Returns camera translation matrix for the given camera. Returns NULL if camera is not being used by the algorithm.

Parameters:

| cameraIndex | Index of the camera |

Returns:

Camera translation matrix

void Neurotec.SentiSculpt.NSfmPointCloud.CopyBaseImages (String rootFolder)

CopyBaseImages Copies base images used for point cloud generation to a specified folder.

Parameters:

| rootFolder | Output folder |

void Neurotec.SentiSculpt.NSfmPointCloud.GenerateOutputForMeshLab (String cameraPositionsFilePath, String baseImageNamesFilePath)

GenerateOutputForMeshLab Generates a file structure which can be used as an input for Meshlab texture mapping.

Parameters:

| cameraPositionsFilePath | File name to write camera positions |
| baseImageNamesFilePath | File name to write base image names |

void Neurotec.SentiSculpt.NSfmPointCloud.GenerateOutputForPmvs (String rootFolder)

GenerateOutputForPMVS Generates a file structure which can be used as an input for PMVS2 dense point cloud generator.

Parameters:

| rootFolder | Output folder |

int Neurotec.SentiSculpt.NSfmPointCloud.GetCameraCount ()

GetCameraCount Returns total camera count used.

Returns:

Camera count

NSfmMatrix<double> Neurotec.SentiSculpt.NSfmPointCloud.GetCameraPositionMatrix (int index)
GetCameraPositionMatrix Returns camera position matrix for the given camera. Returns NULL if camera is not being used by the algorithm.

**Parameters:**

| cameraIndex | Index of the camera |

**Returns:**

Camera position matrix

`NSfmMatrix<double> Neurotec.SentiSculpt.NSfmPointCloud.GetCameraRotationMatrix(int index)`

GetCameraRotationMatrix Returns camera rotation matrix for the given camera. Returns NULL if camera is not being used by the algorithm.

**Parameters:**

| cameraIndex | Index of the camera |

**Returns:**

Camera rotation matrix

`int Neurotec.SentiSculpt.NSfmPointCloud.GetMeshCount()`

GetMeshCount Returns mesh count.

**Returns:**

Mesh count

`void Neurotec.SentiSculpt.NSfmPointCloud.ImportDensePointsFromFile(String fileName)`

ImportDensePointsFromFile Loads dense pointcloud data from a file.

**Parameters:**

| fileName | Filename to load points |

`void Neurotec.SentiSculpt.NSfmPointCloud.SaveBaseImageNames(String fileName)`

SaveBaseImageNames Save base image names used for point cloud generation.

**Parameters:**

| fileName | Output file name |

`void Neurotec.SentiSculpt.NSfmPointCloud.SaveCameraPositions(String fileName)`

SaveCameraPositions Save camera positions used for point cloud generation.

**Parameters:**

| fileName | Output file name |
void Neurotec.SentiSculpt.NSfmPointCloud.SaveDenseCloudToFile (NSfmPointCloudFormat format, String fileName)

SaveCloudToFile Saves dense point cloud to a file.

**Parameters:**

<table>
<thead>
<tr>
<th>format</th>
<th>File format to be used</th>
</tr>
</thead>
<tbody>
<tr>
<td>fileName</td>
<td>Save file name</td>
</tr>
</tbody>
</table>

void Neurotec.SentiSculpt.NSfmPointCloud.SaveMeshToFile (int meshIndex, NSfmMeshExportFormat format, String fileName)

SaveMeshToFile Save reconstructed mesh(es) to file(s)

**Parameters:**

<table>
<thead>
<tr>
<th>format</th>
<th>Output format</th>
</tr>
</thead>
<tbody>
<tr>
<td>fileName</td>
<td>Output file name</td>
</tr>
</tbody>
</table>

void Neurotec.SentiSculpt.NSfmPointCloud.SaveProjectionMatrices (String rootFolder)

SaveProjectionMatrices Save projection matrices used for point cloud generation to a file.

**Parameters:**

| rootFolder | Output folder          |

void Neurotec.SentiSculpt.NSfmPointCloud.SaveVisibilityData (String rootFolder)

SaveVisibilityData Save visibility information to file vis.dat

**Parameters:**

| rootFolder | Output folder          |

void Neurotec.SentiSculpt.NSfmPointCloud.SaveSparseCloudToFile (NSfmPointCloudFormat format, String fileName)

SaveCloudToFile Saves sparse point cloud to a file.

**Parameters:**

<table>
<thead>
<tr>
<th>format</th>
<th>File format to be used</th>
</tr>
</thead>
<tbody>
<tr>
<td>fileName</td>
<td>Save file name</td>
</tr>
</tbody>
</table>
Neurotec.SentiSculpt.NSfmTaskProgress Class Reference

Inherits Neurotec.SentiSculpt.NSfmObject.

**Public Member Functions**

- **NSfmTaskProgress ()**
- **NSfmTaskProgress (HNSfmTaskProgress safeHandle)**
- **SfmTaskResult GetCurrentProgress (ref SfmTask currentTask, ref int progress)**

GetCurrentProgress Returns current task and its progress.

**Additional Inherited Members**

**Constructor & Destructor Documentation**

- **Neurotec.SentiSculpt.NSfmTaskProgress.NSfmTaskProgress ()**
- **Neurotec.SentiSculpt.NSfmTaskProgress.NSfmTaskProgress (HNSfmTaskProgress safeHandle)**

**Member Function Documentation**

- **SfmTaskResult Neurotec.SentiSculpt.NSfmTaskProgress.GetCurrentProgress (ref SfmTask currentTask, ref int progress)**

GetCurrentProgress Returns current task and its progress.

**Returns:**

SfmTaskResult Indicates success or failures

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SfmTask</td>
<td>current task</td>
</tr>
<tr>
<td>int</td>
<td>progress of current task</td>
</tr>
</tbody>
</table>
Neurotec.SentiSculpt.NStructureFromMotion Class Reference

Point cloud implementation. Inherits Neurotec.SentiSculpt.NSfmObject.  

Public Member Functions

NStructureFromMotion ()

NStructureFromMotion (HNStructureFromMotion safeHandle)

void SetTrackLength (int trackLength)

SetTrackLength set the track length.

int GetTrackLength ()

GetTrackLength Returns the track length.

void SetMinTriangularThreshold (int minTriangularThreshold)

SetMinTriangularThreshold Set the minimum triangulation threshold.

int GetMinTriangularThreshold ()

GetMinTriangularThreshold Returns the minimum triangulation threshold.

void SetSequenceLength (int sequenceLength)

SetSequenceLength Set the sequence length. If Sequence is on, image set will be divided in to mini batches by size of sequence length.

int GetSequenceLength ()

GetSequenceLength Returns the sequence length.

void SetSequenceOnOff (NBool isOn)

SetSequenceOnOff Set the sequence on or off. If off, sequence length will be ignored and images will be compared by pairs.

NBool GetSequenceOnOff ()

GetSequenceOnOff Returns current value for SequenceOnOff.

void SetDensePointGenerationOptions (NSfmDensePointGenerationOptions options)

SetDensePointGenerationOptions Set dense point cloud generation options.

NSfmDensePointGenerationOptions GetDensePointGenerationOptions ()

GetDensePointGenerationOptions Returns point cloud generation options.

NSfmPointCloud GetPointCloud ()

GetPointCloud Returns the point cloud generated using candidate images.

SfmTaskResult GeneratePointCloudFromFolder (String hFolderName)

NSfmGeneratePointCloudFromFolder Generate point cloud from contents of an image folder synchronously.

SfmTaskResult GeneratePointCloudFromFile (String hFileName)

NSfmGeneratePointCloudFromFile Generate point cloud from contents of an text file containing image names synchronously.

SfmTaskResult GeneratePointCloudFromFolderAsync (SfmCallback callback, String hFolderName)

NSfmGeneratePointCloudFromFolderAsync Generate point cloud from contents of an image folder asynchronously.

SfmTaskResult GeneratePointCloudFromFileAsync (SfmCallback callback, String hFileName)

NSfmGeneratePointCloudFromFileAsync Generate point cloud from contents of an text file containing image names asynchronously.

NSfmTaskProgress GetAsyncTaskProgress ()
GetAsyncTaskProgress Returns progress of the asynchronous task.

Additional Inherited Members

Detailed Description
Point cloud implementation.

Author:
Neurotechnology

Date:
18 December 2014

Constructor & Destructor Documentation

Neurotec.SentiSculpt.NStructureFromMotion.NStructureFromMotion ()

NStructureFromMotion Constructor.

Neurotec.SentiSculpt.NStructureFromMotion.NStructureFromMotion (HNStructureFromMotion safeHandle)

NStructureFromMotion Constructor.

Parameters:

| safeHandle | Underlying safe handle instance |

Member Function Documentation

SfmTaskResult Neurotec.SentiSculpt.NStructureFromMotion.GeneratePointCloudFromFile (String hFileName)

NSfmGeneratePointCloudFromFile Generate point cloud from contents of an text file containing image names synchronously.

Returns:
SfmTaskResult Point cloud generation result

Parameters:

| hFileName | File name containing image names |

SfmTaskResult Neurotec.SentiSculpt.NStructureFromMotion.GeneratePointCloudFromFileAsync (SfmCallback callback, String hFileName)

NSfmGeneratePointCloudFromFileAsync Generate point cloud from contents of an text file containing image names asynchronously.

Returns:
Success of async execution
Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>callback</td>
<td>Callback will be executed after the point cloud generation finished</td>
</tr>
<tr>
<td>fileName</td>
<td>File name containing image names</td>
</tr>
</tbody>
</table>

**SfmTaskResult Neurotec.SentiSculpt.NStructureFromMotion.GeneratePointCloudFromFolder (String hFolderName)**

NSfmGeneratePointCloudFromFolder Generate point cloud from contents of an image folder synchronously.

**Returns:**

SfmTaskResult Point cloud generation result

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hFolderName</td>
<td>Image folder name</td>
</tr>
</tbody>
</table>

**SfmTaskResult Neurotec.SentiSculpt.NStructureFromMotion.GeneratePointCloudFromFolderAsync (SfmCallback callback, String hFolderName)**

NSfmGeneratePointCloudFromFolderAsync Generate point cloud from contents of an image folder asynchronously.

**Returns:**

Success of async execution

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>callback</td>
<td>Callback will be executed after the point cloud generation finished</td>
</tr>
<tr>
<td>hFolderName</td>
<td>Image folder name</td>
</tr>
</tbody>
</table>

**NSfmTaskProgress Neurotec.SentiSculpt.NStructureFromMotion.GetAsyncTaskProgress ()**

GetAsyncTaskProgress Returns progress of the asynchronous task.

**Returns:**

SfmTaskProgress Task progress

**NSfmDensePointGenerationOptions Neurotec.SentiSculpt.NStructureFromMotion.GetDensePointGenerationOptions ()**

GetDensePointGenerationOptions Returns point cloud generation options.

**Returns:**

NSfmDensePointGenerationOptions Options

**int Neurotec.SentiSculpt.NStructureFromMotion.GetMinTriangularThreshold ()**

GetMinTriangularThreshold Returns the minimum triangulation threshold.
Returns:
int Current minimum triangulation threshold value used in the algorithm

NSfmPointCloud Neurotec.SentiSculpt.NStructureFromMotion.GetPointCloud ()

GetPointCloud Returns the point cloud generated using candidate images.

Returns:
NSfmPointCloud Point cloud generated using candidate images

int Neurotec.SentiSculpt.NStructureFromMotion.GetSequenceLength ()

GetSequenceLength Returns the sequence length.

Returns:
int Current sequence length value used in the algorithm

NBool Neurotec.SentiSculpt.NStructureFromMotion.GetSequenceOnOff ()

GetSequenceOnOff Returns current value for SequenceOnOff.

Returns:
NBool Sequence length is ignored or not

int Neurotec.SentiSculpt.NStructureFromMotion.GetTrackLength ()

GetTrackLength Returns the track length.

Returns:
int Current track length value used in the algorithm

void Neurotec.SentiSculpt.NStructureFromMotion.SetDensePointGenerationOptions (NSfmDensePointGenerationOptions options)

SetDensePointGenerationOptions Set dense point cloud generation options.

Parameters:

| options | Option values to be set |

void Neurotec.SentiSculpt.NStructureFromMotion.SetMinTriangularThreshold (int minTriangularThreshold)

SetMinTriangularThreshold Set the minimum triangulation threshold.

Returns:
NResult Indicates success or failures
**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>minTriangularThreshold</td>
<td>Minimum triangulation threshold to be used in the algorithm</td>
</tr>
</tbody>
</table>

**void Neurotec.SentiSculpt.NStructureFromMotion.SetSequenceLength (int sequenceLength)**

SetSequenceLength Set the sequence length. If Sequence is on, image set will be divided into mini batches by size of sequence length.

**Returns:**

NResult Indicates success or failures

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sequenceLength</td>
<td>Sequence length to be used in the algorithm</td>
</tr>
</tbody>
</table>

**void Neurotec.SentiSculpt.NStructureFromMotion.SetSequenceOnOff (NBool isOn)**

SetSequenceOnOff Set the sequence on or off. If off, sequence length will be ignored and images will be compared by pairs.

**Returns:**

NResult Indicates success or failures

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>isOn</td>
<td>Sequence length is ignored or not</td>
</tr>
</tbody>
</table>

**void Neurotec.SentiSculpt.NStructureFromMotion.SetTrackLength (int trackLength)**

SetTrackLength set the track length.

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>trackLength</td>
<td>track length to be used in the algorithm</td>
</tr>
</tbody>
</table>

End.