

## Application Note

# CVBZividLib

Version 1.0.0 vom 23.10.2019

Autor: Simon Pototzky

## A ClassLibrary to convert Data from the Zivid to CVB

By capturing a frame with the Zivid sensor using the Zivid SDK you will receive Data coded in Zivids own variables. To use them in CVB they need to be converted. This is what the functions in the CVBZividLib.dll make possible.

# Copyright

**Copyright © 2018 STEMMER IMAGING AG, Puchheim.**

Technische Änderungen und Fehler vorbehalten.

STEMMER IMAGING, Common Vision Blox, Windows, Visual Basic, Visual C++, C++Builder, Visual Studio.net, Visual C#, Delphi sind eingetragene Warenzeichen.

Alle Rechte an diesem Dokument sind das Eigentum der STEMMER IMAGING AG, Puchheim/Deutschland. Dieses Dokument oder Teile daraus dürfen ohne vorherige schriftliche Zustimmung der Firma STEMMER IMAGING AG in keiner Weise reproduziert oder vervielfältigt, weder in gedruckter, elektronischer, fotografischer oder sonstiger Form, noch in irgendeine andere Sprache übersetzt werden.

# Inhalt

## Inhaltsverzeichnis

1. Declarations .....	4
2. Structs .....	4
3. Functions.....	4
4. Extensions.....	5
5. Examples for usage .....	6

# CVBZividLib

## 1. Purpose for usage

As there has not been an option to work with Zivid Frames in CVB now and there has not been a Toolkit from Zivid to work on Pointclouds and Images, we now have the option to get all the data captured by a Zivid sensor to CVB. To evaluate a Pointcloud or 2D-images, CVB delivers a huge variety of options. So this is quite a logical to combine both.

To use to be able to use the strength of both combined, the CVBZividLib is the connector. To acquire frames with the Zivid sensor you need to use the Zivid SDK. As you do have a Zivid frame acquired you can transform the data to CVB conform formats by using the CVBZividLib. To do further work on the data, you are now able to use all the available tools from the CVB libraries.

## 2. Declarations

- `zividFrame`: Is a frame captured by the Zivid sensor
- `zividArray`: Is an 3d-array which can be calculated from a `zividFrame`
- `CVBPointCloud`: is a `Stemmer.Cvb.PointCloud`
- `Point3Dcf`: `Stemmer.Cvb.Point3Dcf`
- `colorImage`: `Stemmer.Cvb.Image` with 3 planes
- `depthImage`: `Stemmer.Cvb.Image` with 1 plane
- `Rows`: rows of an Image/Array
- `Cols`: Columns of an Image/Array

## 3. Structs

- `CVBZividObject`
  - Get/Set `Stemmer.Cvb.Pointcloud`
  - Get/Set `Stemmer.Cvb.Image`
  - Get/Set `Stemmer.Cvb.Image`
  - Constructor: `CVBZividObject(CVBPointCloud pointCloud1, Image colorImage1, Image depthImage1)`

## 4. Functions

- `CVBPointCloud ZividFrame2CVBPointcloud(Frame zividFrame)`

# CVBZividLib

- `float[,,]` `ZividPointcloud2ZividArray(Frame zividFrame)`
- `Point3Dcf[]` `ZividArray2CVBPoints(float[,,] zividArray)`  
Returns an array containing `Stemmer.Cvb.Point3Dcf` 3d-points.
- `CVBPointCloud` `ZividArray2CVBPointcloud(float[,,] zividArray)`
- `Image` `GetCVBcolorImageFromZividFrame(Frame zividFrame)`
- `CVBZividObject` `GetCVBcolorImageAndCVBPointcloudFromZividFrame(Frame zividFrame)`  
Returns an `CVBZividObject` with a `Pointcloud` and an `colorImage`. `intensityImage = null`.
- `Image` `GetCVBdepthImageFromZividFrame(Frame zividFrame)`
- `CVBZividObject` `GetCVBImageAndCVBPointcloudAnddepthImageFromZividFrame(Frame zividFrame)`  
Returns an `CVBZividObject` with a `Pointcloud`, a `colorImage` and an `intensityImage`.
- `CVBZividObject` `GetCVBdepthImageAndCVBPointcloudFromZividFrame(Frame zividFrame)`  
Returns an `CVBZividObject` with a `Pointcloud`, an `intensityImage` and a `colorImage = null`.
- `CVBPointCloud` `CropNullpoints(SparsePointCloud sparsePointcloudWithNulls, int rows, int cols)`  
Crops all the NAN-Points in every function returning a `Pointcloud`.

## 5. Extensions

- `CVBPointCloud` `ToCVBPointCloud(this Frame zividFrame)`
- `float[,,]` `ToZividArray(this Frame zividFrame)`
- `Image` `GetCVBcolorImage(this Frame zividFrame)`
- `Image` `GetCVBdepthImage(this Frame zividFrame)`
- `CVBZividObject` `GetCVBdepthImageAndCVBPointcloud(this Frame zividFrame)`
- `CVBZividObject` `GetCVBcolorImageAndCVBPointcloudAndCVBdepthImage(this Frame zividFrame)`
- `CVBZividObject` `GetCVBcolorImageAndCVBPointcloud(this Frame zividFrame)`

# CVBZividLib

## 6. Examples for usage

This example shows quickly how to connect to the Zivid sensor using the Zivid SDK and how to get the acquired pointcloud into CVB. The last step shows that after the conversion you can use CVB functions to work with the pointcloud from the zivid sensor. The pointcloud is written to the harddrive.

```
static void Main()
{
    //Initialize Components
    var zivid = new Zivid.NET.Application();
    Zivid.NET.Frame zividFrame = new Zivid.NET.Frame();
    Stemmer.Cvb.PointCloud cvbPointCloud = null;

    //Connect to Zivid sensor
    var camera = zivid.ConnectCamera();

    //Capture Frame
    zividFrame = camera.Capture();

    //Convert Zivid Pointcloud to CVB Pointcloud
    cvbPointCloud = zividFrame.ToCVBPointCloud();

    //Save CVB Pointcloud
    Stemmer.Cvb.PointCloudExtensions.Save(cvbPointCloud, "cvbPointCloud.ply");
}
```