

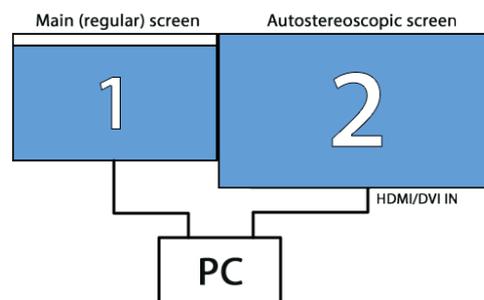
# Autostereoscopic Display Tool - User Manual

The Autostereoscopic Display Tool (hereinafter application) is designed to create and display interlaced multiview images on Philips BDL4251VS autostereoscopic screen.

## Display Setup

In order for the application to work properly, the autostereoscopic screen has to be connected directly to the computer (via HDMI) and set as a secondary monitor positioned to the left of the main one with top edges aligned. Check the picture below for reference.

The desktop has to be set to extended mode, not to screen mirroring or only to display on the second (autostereoscopic) display.



Also make sure the screen format (aspect ratio) is set to “full” instead of other “zoom” setting. In that way the screen displays true 1:1 pixel reproduction of the source signal with no overscan.

If the output images look incorrectly with too much crosstalk, check that the signal delivered to the screen is in YCbCr format and if that is the case, try to set the output format on the computer to RGB.

## Instalation

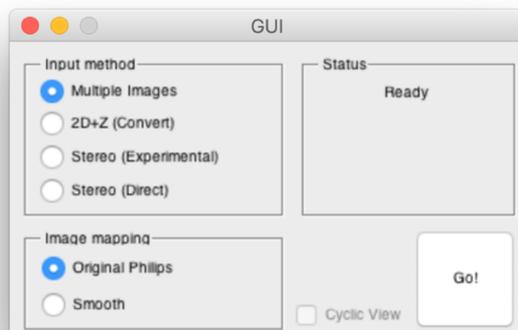
Both Windows and OS X versions of the software have been compiled into an executable applications independent on Matlab installation but as the application is written in Matlab language, it still requires a correct version of Matlab Runtime in order to work properly.

The runtime is installed with the application itself, downloaded from the internet during the installation process. The installation is pretty straightforward and does not require any unusual steps. It may, however, take some time, depending on your internet connection.

The default installation path on Windows is */Program Files/Autstereoscopic\_Display\_Tool/* and on OS X, the application is installed directly to */Applications/*.

## Using the application

The main window of the application is divided into four sections. The “Input method” section, “Image mapping” section, status information area and a program-launching button.



Input method selection allows the user to set the format used as an image source:

### Multiple images

Multiview images format used for 6-28 individual image files, each representing a different viewing angle of the displayed scene.

### 2D+Z (Convert)

2D plus depth image format consisting of two separate image files, one containing RGB “middle” view of the displayed scene and the other its grayscale depth map. This format is converted into a set of 28 artificial multiview images.

### Stereo (Experimental)

A standard stereo pair format using two image files, “left” and “right”. This format is converted first to 2D+Z format and through that into 28 multiview images, effectively creating 28 new views of the displayed scene.

### Stereo (Direct)

Uses the same input format as the experimental method but displays it into two wide viewing zones, creating a two-view autostereoscopic image. This displaying method is not suitable for regular image viewing as it produces several artifacts.

The Image mapping section selects the interlacing method used for the creation of the final output image:

### Original Philips

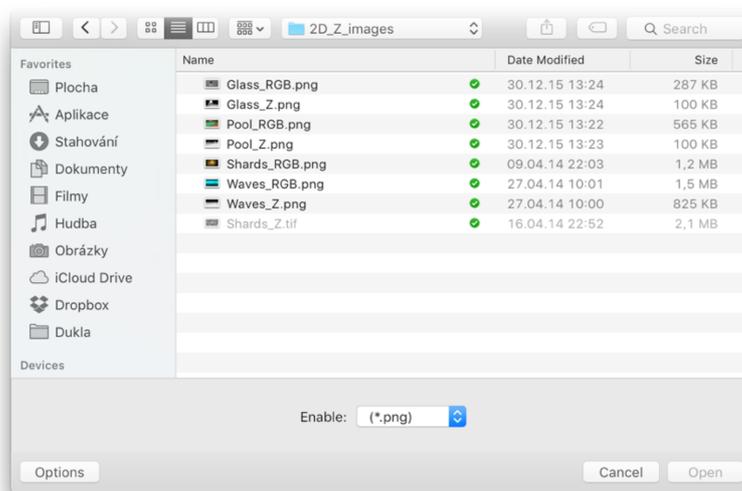
A view-mapping method using only 15 source images to reduce the cross-talk effect and other unpleasant artifacts.

## Smooth

A direct view-mapping method using the full range of 28 source images resulting in smoother motion parallax at the cost of additional undesirable effects (grater cross-talk, right-to-left image transition ...). The “**Cyclic View**” checkbox modifies the method, using only 14 images to suppress the image “flipping” effect.

With the parameters set, launch the process of creating the interlaced output image with the press of the “Go!” button. A new dialog window opens prompting the you to select the source image files (either all at once in case of “Multiple Images” format or one by one in the other two cases).

The application supports files in .jpg, .png, .tif and .bmp formats.



After the source files are selected, press the Open button and the creation of the output image starts. The current progress information is displayed in the main window “Status” area.

(Note: The computation may take some time, percentage information in the status area is only approximate.)

After the creation of the interlaced image is done, it is displayed in a fullscreen-sized window and a dialog box opens, giving the option to save the generated image into a .png file.



After the image is saved or the dialog box is closed either by selecting “No” or by closing its window, the application is ready to process new images.

## Troubleshooting

**Problem:** The application cannot be launched, returning a runtime error.

**Solution:** Try reinstalling the runtime. If that does not solve the problem, try running the application straight from Matlab using the GUI command.

**Problem:** The application takes long time to launch and to deliver images.

**Solution:** Wait. For reasons most likely connected to the Matlab runtime, the application is quit quite slow (especially under Windows OS). Using the GUI command straight in the main Matlab software might result in slightly faster response.

**Problem:** The application reports “Finishing up” but nothing is happening.

**Solution:** The finishing part of computation takes a long time on Windows machines. Look at memory usage of the application and if it uses several thousands of MB of RAM, the computation is still running. If the memory usage is lower, and it is possible to change the settings in the main application window, an error occurred. If problems persist, launch the application from Matlab and check the command window if it returns the error report.

**Problem:** The image is generated but appears blocky.

**Solution:** The image is probably scaled to the size of its window, which is smaller than required. Check that all OS elements (taskbar, dock, menubar etc.) are hidden on the secondary display and the image window covers the full area of the screen. If not, try to maximize it.

**Problem:** The program returns error when generating depth map from stereo pair.

**Solution:** If you are running the application straight from Matlab environment and it reports unknown function, you might be missing some of the required toolboxes. Try running the standalone application as the necessary code should be available as part of the Matlab Runtime. Another workaround is to generate depth map using a third party software such as DMAG5<sup>1</sup> and then use it as 2D+Z input. This solution should also give better results as more advanced algorithm is used.

---

<sup>1</sup> <http://3dstereophoto.blogspot.cz/p/software.html>