**USER GUIDE**

«Development of a subsystem for creating a depth map for a stereo vision system»

Batrakova Elena Sergeevna

Batonova Oksana Urievna

Content

[Content 2](#_Toc132375424)

[Annotation 3](#_Toc132375425)

[Introduction 4](#_Toc132375426)

[Instruction 6](#_Toc132375427)

[Glossary 13](#_Toc132375428)

[Subject index 15](#_Toc132375429)

Annotation

This guide contains detailed instructions on using the author's algorithm for automatic database generation in the Blender environment. This algorithm provides a fast and high-quality solution to the main problem when working with deep learning, namely, a high-quality base and realistic data that are close to real measurements. This will be a good starting point for working with depth maps in related research, and can also serve as an excellent learning resource for students who would like to immerse themselves in the field of stereo vision and modeling. A high-quality dataset is also always a time saver for the developer, which means using fewer resources for the same amount of work that was done before.

It is also worth noting that any researchers who are passionate about the topic can use the results, since the code will be used in the public domain. Therefore, it can be used as a basis for further work and give an idea in order to modify the code for other needs.

Introduction

Recently, the use of synthetic data in training models has increased significantly due to their increased efficiency and speed. Deep learning architectures require a large amount of training data, and while large-scale visual data collection is relatively easy to implement, annotating the collected data is time consuming. While only a small dataset is needed for fine-tuning when using pretrained models, labeling just a few hundred or a few thousand images to segment objects or instances is still a laborious task if done manually.

The choice of specific software will be justified next. Blender was chosen as the main tool because it is a popular open-source software that can provide a number of development benefits. Some of them:

a) Synthetic image generation: Blender can create 3D models of objects and scenes and generate synthetic images with a wide range of camera angles and object shapes, allowing the subsystem to be trained.

b) Realistic Lighting and Materials Simulation: Simulation of various lighting conditions, material properties and environmental factors helps in creating realistic images.

c) Economical: This is cost effective compared to using real world images, which may require expensive equipment, logistics and labor. Using Blender, the research team saves resources and distributes them to other important aspects of the project.

d) Efficient workflow: A user-friendly interface and a complete set of tools simplify the workflow, in particular, a plug-in mechanism to extend the built-in functionality.

The Python API for Blender allows developers to access and control Blender functionality using the Python programming language, and gives researchers the ability to automate tasks, customize workflows, and extend Blender's functionality by creating custom scripts and add-ons.

The package consists of a set of modules and functions that provide access to Blender's internal data structures such as objects, materials, textures, and animations for creating, modifying, and deleting objects, as well as interacting with its interface. For example, a script can be written for randomly placed objects in the scene and change their size, orientation, and material properties. You can write another script to automatically create camera viewpoints from a given range of angles and distances.

The Python API allows you to automate repetitive tasks such as setting up scenes, importing and exporting files, and generating synthetic images. This automation saves time and reduces the chance of errors. In the context of development, the API is used to automate the generation of training data and the export of files.

Instruction

This section provides a step-by-step operation plan for the developed subsystem for constructing an image depth map for a stereo vision system.

1. Open Blender on your computer. If you have not come across this software before, you need to download and install it on your device. Use the official site https://www.blender.org/download/ (Fig. 1). Blender is free and open source, making it available to everyone. This tutorial covers the latest version of Blender 3.5.

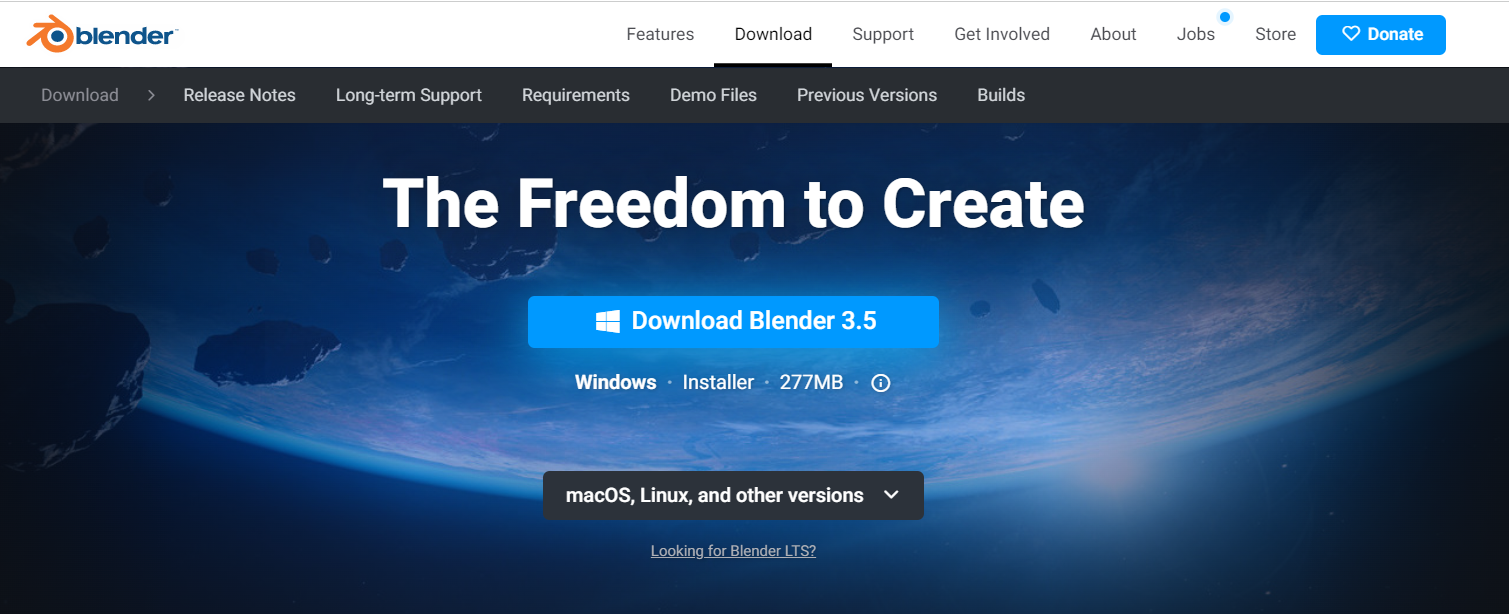


Fig. 1. Official link to download Blender.

1. After installing and successfully launching Blender, let's turn to the settings. In the control panel, in the upper left corner, select "Edit", in the list that opens, select the "Preferences" item (Fig. 2).

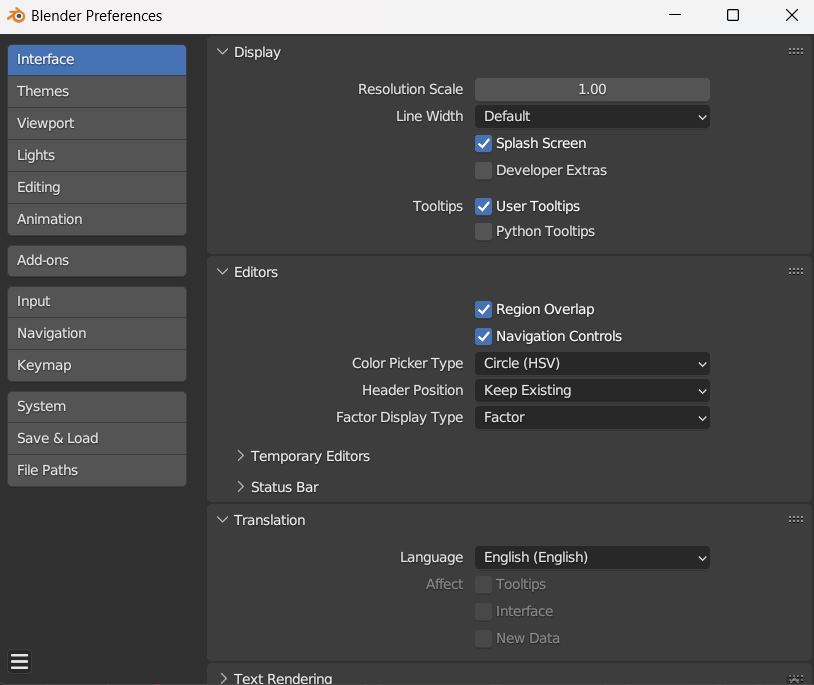


Fig. 2. Preferences pop-up window.

1. In the window that opens, select the "Safe & load" section from the list on the left. Find the item "Auto Run Python Scripts" and activate the checkbox on the left side (Fig. 3).

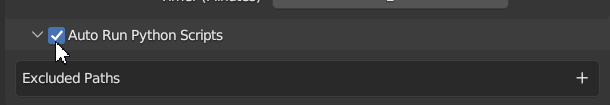


Fig. 3. Configuring automatic script execution.

1. Next, let's move on to installing the archive. To do this, repeat the previously done path: "Edit" >>> "Preferences". Next, click on the "Add-ons" section, and then on the "Install" button in the upper right corner. You will see a Documents section and a list of files stored on your device. Among them we will find the necessary zip-archive. Type in the search: “stereovision\_dataset.zip” (Fig. 4), then click the blue “Install add-on” button in the lower right corner.

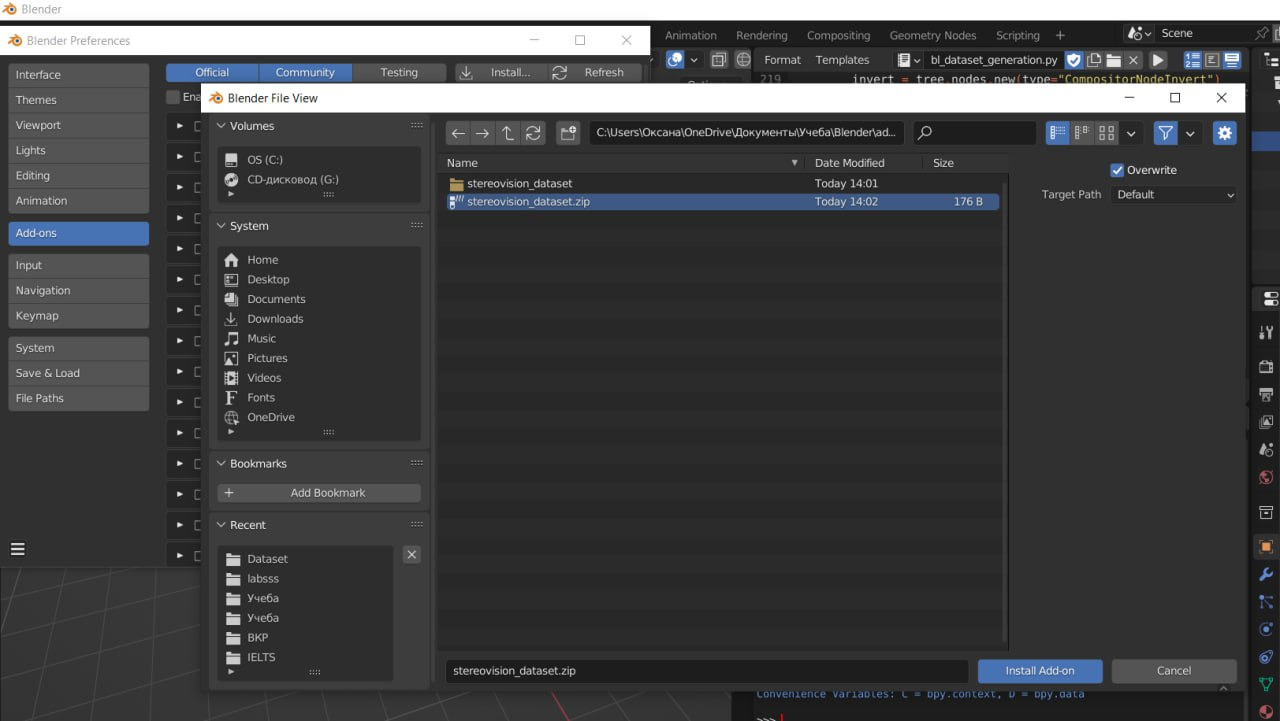


Fig. 4. Loading the script into Blender.

1. After loading the script, you can see it in the list of all addons (Fig. 5), if you do not see this addon, then close Blender and then start the application again.

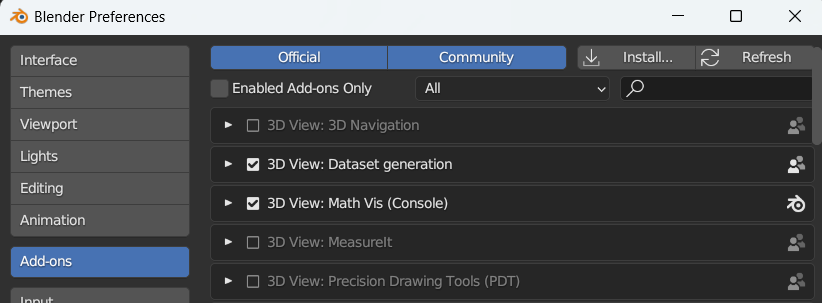


Fig. 5. List of Addons in Blender

1. To use the downloaded addon, use the arrow in the upper right corner of the workspace (Fig. 6).

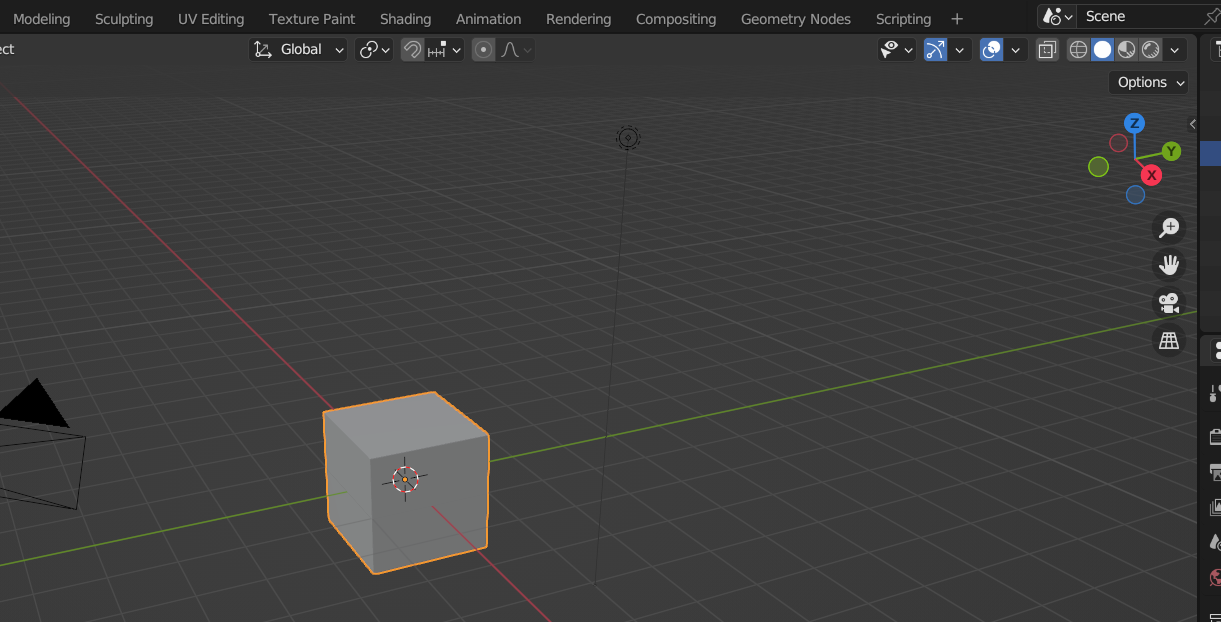


Fig. 6. Blender workspace.

1. In the window that appears, select the "Datasets" tab (Fig. 7).

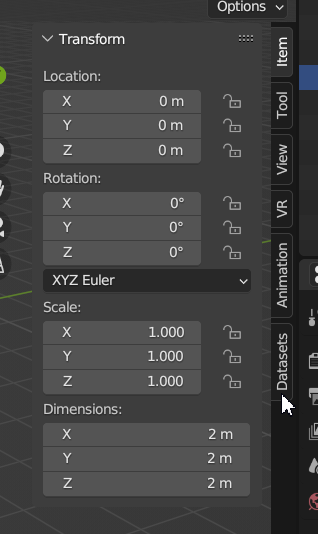


Fig. 7. Window for working with addons.

1. let's proceed directly to creating the dataset. To do this, fill in all the fields in the "Dataset Generation" window that appears. If some of the fields are not filled in, the "Generate" button will remain unclickable and the inscription on it will be painted in a dull gray color (Fig. 8).

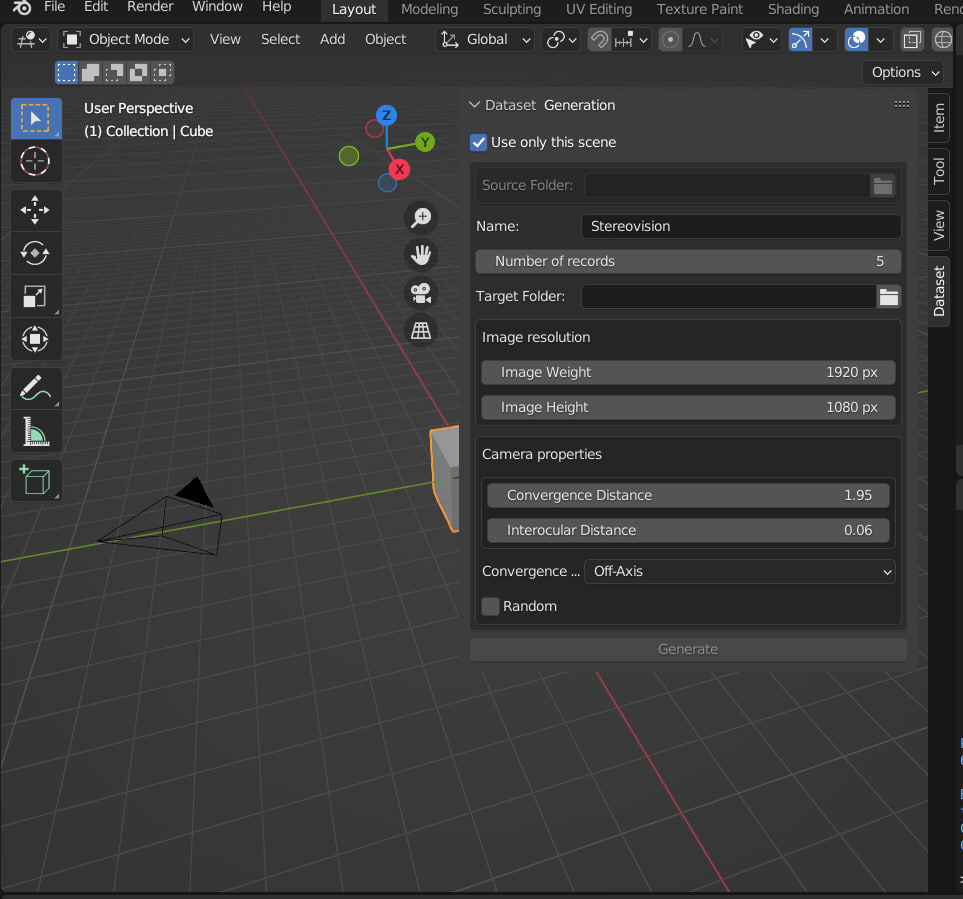


Fig. 8. Filling in the parameters in the "Dataset Generation" window.

1. Let's consider each parameter in more detail. If you click and check the box next to the “Use only this scene” item (Fig. 9), you will only need to fill in two fields: “Name” (up to 30 characters) and “Target Folder”, namely the name of the dataset and specify the folder where the dataset will be saved. You may notice that after filling in these fields, the “Generate” button became active, which means that the dataset can be generated.

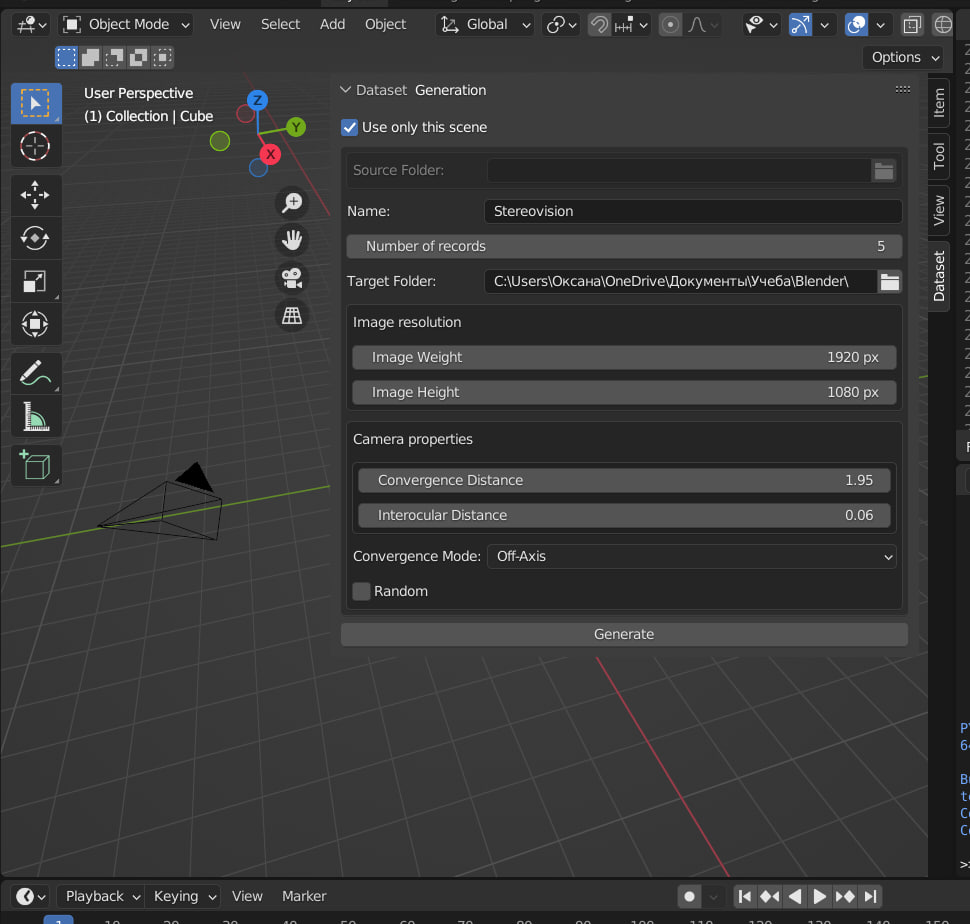


Fig. 9. Filling out form fields.

1. Unchecking the box "Use only this scene" implies that additional scenes will be used, which means that they must be loaded. This can be done by specifying the path to the files in the "Source Folder" column (Fig. 10). Please note that the extension of the files accepted for download is written as .blend.

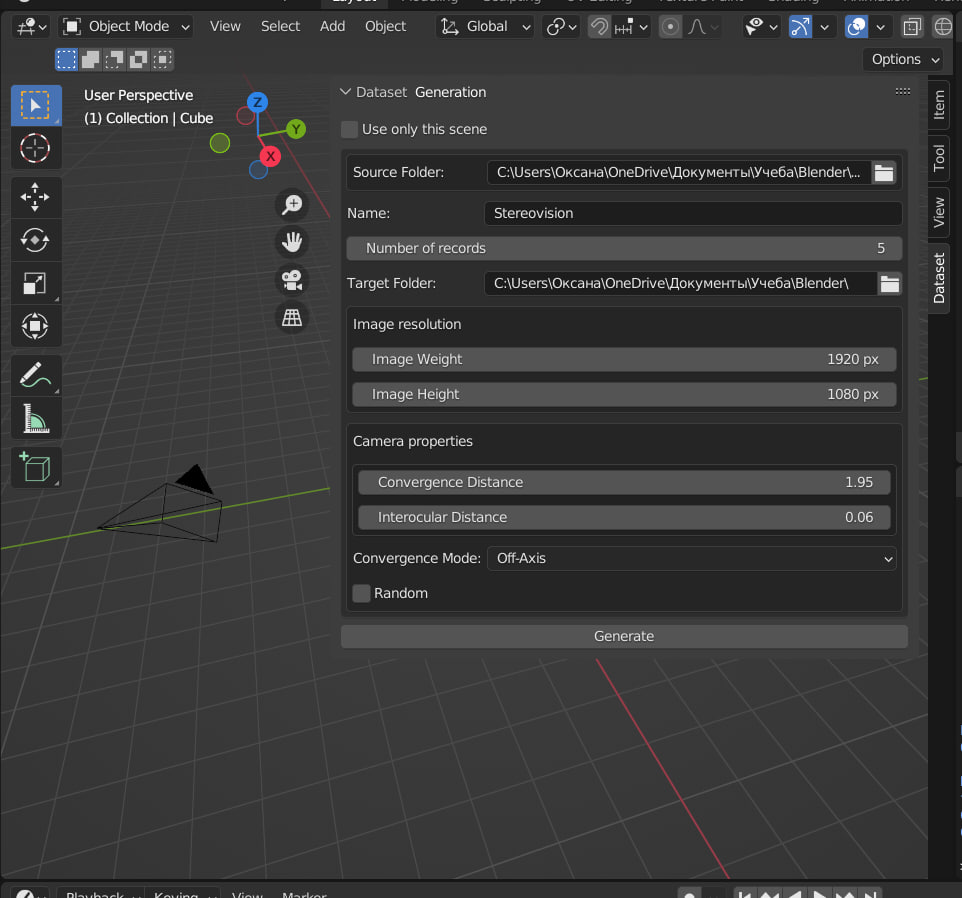


Fig. 10. Selecting additional scenes.

1. At the bottom of the window, you can see the "Random" item (Fig. 11). It is responsible for the settings of the cameras of the stereo vision system, listed in the "Camera properties" section. If you check the box, the values ​​are set randomly, and if you uncheck it, the parameters can be configured manually.

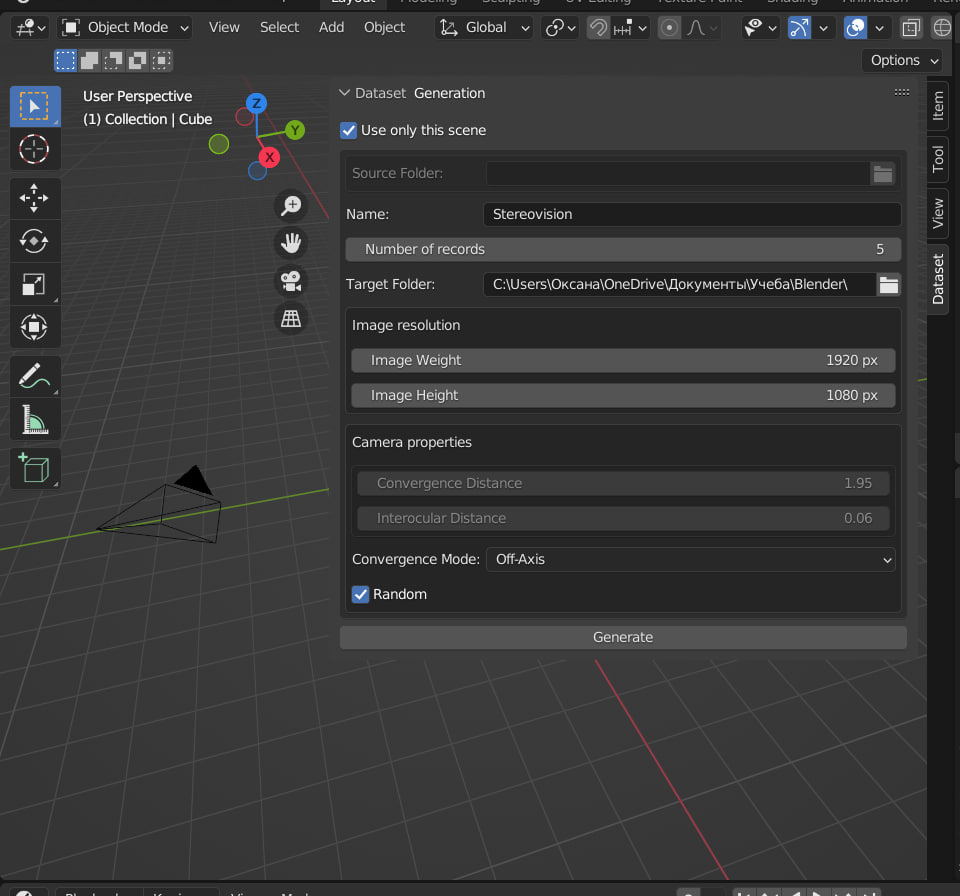


Fig. 11. Random setting of camera parameters.

1. Pay attention to the preset parameters in the "Image resolution" and "Camera properties" sections, which are responsible for the image size and camera settings, respectively. For "Image Weight" it's 1920 px, for "Image Height" it's 1080 px. For "Convergence Distance" it's 1.95, for "Intercocular Distance" it's 1.06. Also, in the “Number of records” section, which is responsible for the number of records in the dataset, the number 5 is initially indicated.
2. Each parameter has its own maximum and minimum values that can be specified. Number of records from 1 to 10.000. Image resolution up to 50,000 pixels. For two distances, the minimum is 1×e^(-5) and 0 respectively. You can enter the parameters after double-clicking on the required line.
3. After generating the dataset, let's look into the folder to evaluate the result (Fig. 12). The "images" folder contains photos of each scene from two angles. Depthmap contains depth maps. "Readme.md" contains instructions for decoding the text accompaniment to the created dataset. Familiarize yourself with it before moving on to the next file. The .txt file has the name you specified earlier in the "Name" column, here it is "Stereovision.txt". It contains textual descriptions of the received depth maps with parameters, coordinates and names of objects on images.

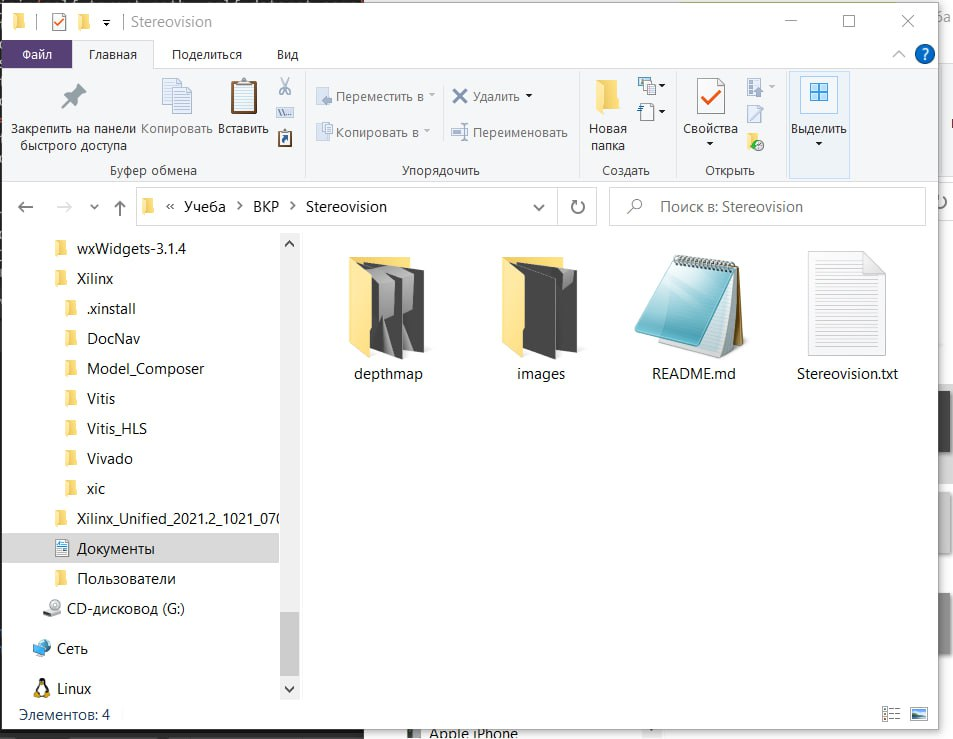


Fig. 12. Received dataset, output files.

Glossary

1. **Blender** - professional free and open source software for creating three-dimensional computer graphics, including tools for modeling, sculpting, animation, simulation, rendering, post-processing and editing video with sound, compositing using "nodes", as well as creating 2D animations .

2. **Python** is a high-level general-purpose programming language with dynamic strong typing and automatic memory management, focused on improving developer productivity, code readability and quality, as well as ensuring the portability of programs written in it.

3. **Python API** (application programming interface) - application programming interface.

4. **Database** - a set of data stored in accordance with the data schema, the manipulation of which is performed in accordance with the rules of data modeling tools.

5. **Deep learning** is a set of machine learning methods based on learning representations, rather than specialized algorithms for specific tasks.

6. **Dataset** is a collection of data. In the case of tabular data, a dataset corresponds to one or more database tables, where each table column corresponds to a different variable and each row corresponds to a record in the dataset.

7. **A depth map** is an image or image channel that contains information regarding the distance of the surfaces of objects in the scene from the viewpoint.

8. **Software** (SW) - a program or set of programs used to control a computer.

9. **Synthetic data** is information created artificially and not as a result of real events. Synthetic data, usually created using algorithms, can be used to validate mathematical models and train machine learning models.

10. **Stereo vision** (stereoscopic vision) - a type of vision in which it is possible to perceive the shape, size and distance to an object, for example, thanks to binocular vision (the number of eyes can be more than 2, as, for example, wasps have two compound eyes and three simple eyes (eye), scorpions - 3-6 pairs of eyes) or other types of vision.

Subject index

Blender, 3, 4, 5, 6

database, 3, 4

dataset, 4, 9, 11

Dataset, 8

deep learning, 3, 4

depth map, 3, 6, 12

Python, 4

Python API, 4, 5

software, 4, 6

stereo vision, 3, 6, 10

Synthetic image generation, 4