Blenderer 2.1.0

Addon for rendering more realistic images and videos. This version 2.1.0 is developed and tested with Blender version 2.82. Visit https://www.blender.org/ to download and install blender before using this addon.

Once the Blender is installed, extract the Blenderer_2_1_0__2020_09_28.zip to *"C:\Users\AhoneSa1\<USERNAME> \Visual Components\4.2\My Commands"* and then launch Visual Components. On the first time when trying to render, the addon might prompt user to locate the installation path of the blender.exe file. By default, it is assumed that blender executable is found at *"C:\Program Files\Blender Foundation\Blender 2.82\blender.exe"*.

Properties in the addon user interface and how to use them

Default

Default tab has the most important properties for rendering. You can simply open the Blenderer addon and hit render. There's a default lighting settings that will light up the scene in the render.

Property	Default	Description
Output file	C:\Users\<USERNAME>\Pictures\ Blenderer\Picture.png	File path to save the rendering result. Warning: If there's an existing file, the file name will get autoincremented number at the end. If the filename ends with a number, that number will be incremented
Samples	8	Number of rendering samples per each frame. Higher value gives better, less noisy, result, but will increase rendering times.
Resolution	"<Custom>"	List of preset resolutions. Use custom for editing X and Y resolution manually. Higher resolution will require longer rendering time.
ResX	1920	Horizontal resolution of the output image
ResY	1080	Vertical resolution of the output image
Page	"Landscape"	Portrait setting will flip horizontal and vertical resolutions in the output image
Lighting Preset	3 Light Studio	Changing this will automatically set a corresponding HDRI map to light the scene. Also depending on the preset, some additional lights

		may be added to the render scene. To create your own lighting presets, see <command folder>\Lights\Presets\presets.xml
Render		Calls addon to convert the scene from Visual Components into Blender and rendering it with current settings.
Reuse Previous Scene	False	Set to True before rendering to quickly re-render same scene again. You may change lighting and re-position camera and components in the scene, but all same nodes must stay in the scene compared to previous render call. (re-rendering animations with dynamic components not fully supported)

Lights

The visual components scene is lit up with two default lights. These lights are not exported to Blender. The render engine will require special light sources that will define their type, color, size and strength. These light sources can be defined in the Visual Components scene with special components that will be converted to actual light sources in the render scene. Use the buttons on the Lights tab to create light components. Light components will be created to a current location of the camera. Light components won't have any effect on the lighting in the Visual Components scene. Light source components won't be visible in the scene. Only the light they emit. Notice that the current Lighting Preset may add some lights to the scene before rendering. Use *NULL* preset to fully control the lighting with your own custom setup.

Property	Default	Description
Add Point Light		Adds a point light to a current location of the camera (view). Point light emits light to all directions.
Add Area Light		Adds an area light to a current location of the camera (view) directed to the same direction as the camera. Area light emits light from one side of the area surface.
Add Spot Light		Adds a spotlight to a current location of the camera (view) directed to the same direction as the camera. Spot light emits light from a point like source to a direction defined by a cone.
Add Sun Light		Adds a sun light to a current location of the camera (view) directed to the same direction as the camera. The sun light emits one directional light. The location of the sun light has no effect. Only the orientation.

Default Strength	500	The strength of the new light in watts when it is added to a scene (strength of the sun light is the value divided by 10 (500W defaults to 50W) Strength value can be later edited as a component property of each light source component.
Add Sparks Emitter		Adds an emitter component to the scene close to the current camera view. Typical use: Attach the emitter to the welding torch and connect a control signal to the emitter. When *Sparks* signal is on the emitter will emit sparks in the rendered scene. This component will work properly only with animations and not so much with stills. By default the sparks are blueish in color and it will dim other lights in the scene down to 10% to highlight the spark effect. The color and dimming rate can be configured as component properties in the sparks emitter component.

Hint: Rotate and navigate camera (view) to different locations around the scene and add light sources. Use e.g. 3 area lights to create a 3-point studio lighting.

Lights are custom components in the Visual Components scene that can be moved around and even attached to moving nodes to record moving lights in animations. Select a light component in the Visual Components scene to access the component properties. The meaning of the properties in the light components are explained below. Not all properties are found in every light source type.

- Size (Point / Spot / Area)
    - Size of the light source. The smaller the size, the sharper the shadows casted from this source.
    - Area light has SizeX and SizeY properties separately.
- Strength (Point / Spot / Area / Sun)
    - Intensity of the light source. Light source strength is usually in tens of thousands except for the sun light where e.g. value 2 is already pretty bright.
- R,G,B (Point / Spot / Area / Sun)
    - Color of the light can be defined as rgb values (e.g. 255, 255, 255 represent perfect white)
- Angle (Spot)
    - Angle of the spot cone can be defined with the Angle property in degrees.
- Falloff (Point / Spot / Area) [NOT SUPPORTED BY THE EEVEE RENDER ENGINE USED BY BLENDERER 2.X.X]
    - In real life, the intensity of light as a function of the distance from the light source follows an inverse square relationship. So the intensity falls quite rapidly in function of distance. This can be modified to more easily light up bigger scenes with less light sources.
    - Options:

- Quadratic: Physically accurate falloff with inverse square relationship
- Linear: Light intensity falls linearly in function of distance
- Constant: Light intensity doesn't depend on the distance (e.g. area and spot lights still emit to certain directions)
  - Sunlight doesn't have falloff option.

Advanced

More detailed and advanced features can be configured on the advanced tab.

Property	Default	Description
CameraFov	40	Camera Field of View defines what angle the camera sees in the scene. Higher value will result to wider-angle image. The view in the Visual Components doesn't fully respect the framing in the result image. *Hint: use low quality settings for quick test rendering to see the result image framing.*
Show Border	False	Enable this and change the CameraFov to see the rendering framing border in Visual Components scene. Helps to frame the image.
Exposure	-1.5	Camera exposure value affects on the image brightness. Smaller values will create darker image and higher value creates brighter image.
Bloom	False	Enables bloom effect around the bright areas in the image.
Bloom Intensity	0.1	The strength of the bloom effect
OpenInBlender	False	When OpenInBlender is enabled and rendering is called on the Default or Animation tab, the converted scene is opened in Blender software. Useful feature for users familiar with the Blender user interface.
RenderDevice	"GPU"	Choose between CPU and GPU rendering. Rendering will be done with CPU even if the GPU is chosen if the GPU rendering is not set up in the Blender User Preferences inside the Blender user interface. Launch standalone Blender instance to edit the user preferences. Once setup is done the render device can be chosen for each render using this renderDevice property. Usually GPU rendering is faster, but CPU rendering may support rendering of larger scenes.

| DepthOfFieldNode	*Null*	Camera depth of field can be focused to a center of a node. If set to *Null* the camera DOF is infinite (i.e. all objects are in focus). *Hint: DepthOfFieldNode can be a moving node to create cool cinematographic effects in animations.*
Depth F-Stop	2.8	Lens F-Stop. Changes the depth of field length. Higher value results in higher focus.
Indirect Lighting	Off	Improves the realism in the scene lighting, but requires longer rendering and scene prep times.
Blender Executable	C:\Program Files\Blender Foundation\Blender 2.82\blender.exe	File path which is used to locate the Blender executable. When rendering the scene, this executable is automatically launched. If the blender executable is not found in this given filepath, a popup dialog will appear for user to locate the executable. The new filepath is stored in the defaults.xml file stored in the command folder and read from that file next time when the addon is launched. If the value is changed manually to this property, the new path is not persisted.
Reset All Properties		Resets all properties on all tabs to hardcoded defaults or to a default defined in the defaults.xml file in the command folder.

Materials

Mode	"Default"	With "Default" mode value the scene materials are converted from Visual Components scene to Blender scene as is. With "Write To Table" mode the scene material values are written to a materials.csv file which is saved to the Blenderer command folder. With "Read From Table" mode the material definitions are read from a materials.csv file stored in the Blenderer command folder. Materials that are not defined in the csv are converted directly from Visual Components scene as is. *Hint: Set Materials property first to Write To Table mode and render the scene (when rendering begins it can be aborted, the file is already generated at this point). Then*

		open the materials.csv file and edit the material definitions and save the file. Then Change the Materials property to Read From Table mode and render again with the materials defined in the csv file. This makes it easier and faster to edit the render materials without editing the materials in the Visual Components layout. See an example materials.csv file below.
TableDelimiter	";"	CSV delimiter for the materials.csv table. *Hint: usually semi-colon in European Windows and comma in American/Asian Windows.*
Robot Materials	Smart Materials	Default: Use materials of the robots as they are in the VC model. Smart Materials: Some shading effects are automatically applied to the materials in the components including robot controller behavior. *If Materials::Mode is set to read from excel this setting is ignored.*

Example materials.csv file defining Black and orange materials and adding custom style to them.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Name	R	G	B	metallic	roughness	bumpiness	clearcoat	clearcoat_rouhr	opacity	style	color_map	normal_map
2	Black	0.2	0.2	0.2	0	0	0	0		1	castmetal|0.1|2.0		
3	orange	1	0.3	0	0	0.9	0.01	0		1	pulver|0.05|0.9		
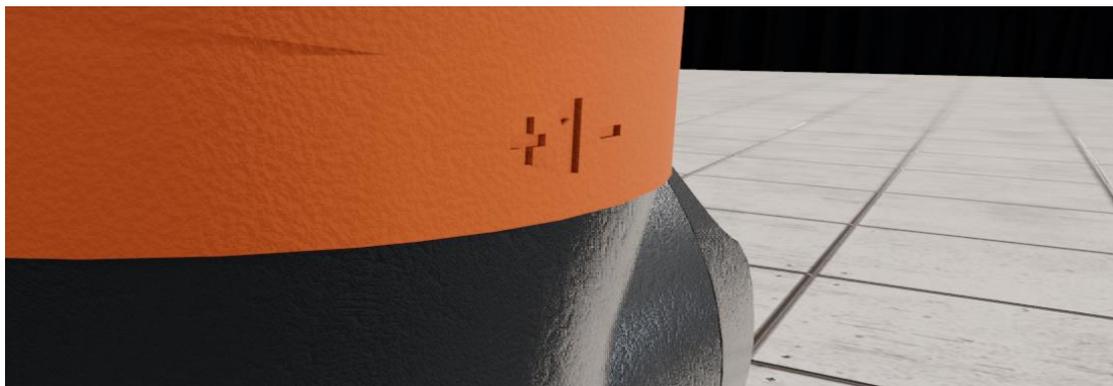
Style column support styles "pulver" and "castmetal". Syntax is "stylename|strength|texturescale"

Additional properties strength and texturescale can be left out and use only the stylename in the column. In the case strength and texturescale are not defined they default to 0.05 and 500.0 respectively.

Animation

Animations can also be rendered. Animations are rendered to a series of image files which are stitched together into a video after all frames have been rendered. Rendering animation requires recording the scene before calling rendering.

Workflow to record and render an animation:

- Check RecAnimation checkbox *on Animation tab*
- Set start time (in seconds)
- Set end time (in seconds)
- Define StepSize (0.04s => 25fps)
- Play simulation until RecAnimation check box is automatically disabled
- Hit "RenderAnimation"
- Animation is rendered frame by frame into image files. File path is defined on the Default tab. Animation frame images will have ascending numeric post fix (e.g. Picture00001.png)

Property	Default	Description
RecAnimation	False	Set this True before running the simulation to record the scene for animation rendering.
Start	5s	Start time of the recording in seconds.
End	10s	End time of the recording in seconds.
Rec Step	0.04s	Defines the animation frame rate. 0.04s => 25fps, 0.02s => 50fps *Warning: recording is executed on OnRender event in Visual Components. Don't change the simulation speed during recording and don't navigate the 3d world.*
FPS	25	Frame rate of the resulting video. This value can be different than what the Rec Step setting suggests. *E.g. Rec Step 0.04 and FPS 50 will result a 50FPS video with the half of the speed of the simulation.*

| Container | MP4 | Choose the video file container format. Choose **MP4** for portability (smaller file) OR choose **AVI** for reliability (better compatibility in some cases). |
| RenderAnimation | | Launch rendering by hitting this button after the recording is done. |

Env

Environment map has two roles. It can show up in the rendered image as the background image and it can be used as a "light source" in the scene. Background can be set to be a plain RGB color or a HDRI environment map image (Google "HDRI map" for examples). Prefer very high resolution and high contrast maps for better results. The example map shipped with the Blenderer addon is quite poor quality both in contrast and in resolution.

Property	Default	Description
Background	Image	Choose between Image or ColorRGB. Depending on the selection either Image or ColorRGB property is shown.
Image	<command folder>\ Lights\Presets\studio_small_01_4k.hdr	Set an environment image map. Most commonly used image formats are supported including *.hdr. Environment image is wrapped around the scene with spherical projection so normal planar images are not applicable.
ColorRGB	Vector(1,1,1)	Color defined as RGB values between 0…1. Default value X1, Y1, Z1 represent white and can be understood as R=255 G=255 B=255
Strength	0.3	Strenth of the light emitted from the map. Higher value lights up the scene more. Setting this to 0 will disable Env map as a light source
Rotation	0	Rotates the HDRI map around Z-axis. Changes light direction in result image.
Visible	True	Map can be used as a light source even if the map is not shown as the background in the output render.

Floor

In a real life light bounces from the floor. So even if the light sources in a room are usually in the ceiling, a lot of light is bounced back from the floor lighting the objects from below. So having a floor in the render scene is quite important for more realistic result. The default floor in the Visual Components scene is not converted to the render scene. Render scene floor can be easily configured on the floor tab.

Property	Default	Description
Visible	True	Disable this if the render scene floor is not needed. *Hint: If the floor already exists in the scene as 3d geometry it is better to disable the render scene floor completely*
BorderSize	2000mm	Floor size is automatically calculated from the scene contents so that the floor covers all the items in the scene. BorderSize defines how much the floor overshoots every direction around the items in the scene.
Style	SlateBrown	Presets of floor styles. Changing this will overwrite the existing ColorTexture and BumpTexture property values. Custom floor style can be defined by simply setting any texture (image) files to the ColorTexture and/or BumpTexture files. Hint: Choose **<ShadowCatcher>** as a style to have an invisible floor that doesn't show up in the render but catches the shadow. It's recommended to hide the Env map and setting its strength to 0 when using ShadowCatcher. Also, it is recommended to create harder lighting with e.g. sun lamps instead of the default lighting presets to make the effect more noticeable.
ColorTexture	<command folder>\ Textures\FlooringSlateBrown.jpg	Image file for texturing the floor
BumpTexture	<command folder>\ Textures\FlooringSlateNormal.jpg	Image file for creating surface shape to the floor. Use normal map type bump map. Rgb values of the each pixel in the normal map defines a XYZ normal vector. Completely blue image (all pixels 0,0,1) represents completely flat surface.
TextureSize	1200mm	Size of the texture tile on the floor geometry. Texture is repeated on the floor. Prefer tileable textures.
ColorRGB	0.8, 0.8, 0.8	RGB color vector for floow color if ColorTexture is not defined or is invalid.