

Readme - Screenshot Helper 1.0.7

Easy capture screenshot, capture any part of the screen or capture image from Unity camera(s).

Features

Captures full-screen image.

Captures part of the screen. Any size, at any screen position.

Captures image from Unity camera(s).

Support touch, including touch-to-capture example.

Save screenshot/cutout texture as JPG, PNG.

Compatible with Pro GIF for saving texture(s) as GIF.

Mobile friendly.

Runtime.

———— Set the callback to receive screenshots ————

Add these code in your script to set the callback for receiving screenshots(Texture2D/Sprite/RenderTexture):

- **Set the callback to receive captured image in Texture2D format:**
ScreenshotHelper.iSetMainOnCapturedCallback((Texture2D texture2d)=>{
 //Your code for handling texture2d:
 //.....
});

or

- **Set the callback to receive captured image in Sprite format:**
ScreenshotHelper.iSetMainOnCapturedCallback((Sprite sprite)=>{
 //Your code for handling sprite:
 //.....
});

or

- **Set the callback to receive captured image in RenderTexture format:**
ScreenshotHelper.iSetMainOnCapturedCallback((RenderTexture renderTexture)=>{
 //Your code for handling renderTexture:
 //.....
});

———— Capture fullscreen / Part of the screen ————

- **Capture fullscreen image:**

[ScreenshotHelper.iCaptureScreen\(onCapturedCallback\);](#)

You can assign an **onCaptured** action in this method to receive the captured image.

- **Capture part of the screen with touch/mouse position & image size:**

[ScreenshotHelper.iCapture\(Input.mousePosition, captureSize, onCapturedCallback\);](#)

or

[ScreenshotHelper.iCapture\(Input.mousePosition, onCapturedCallback\);](#)

You can assign an **onCaptured** action in this method to receive the captured image.

- **Set/Get capture image size(Vector2):**

Set: [ScreenshotHelper.iSetCaptureSize\(new Vector2\(newWidth, newHeight\) \);](#)

Get: [ScreenshotHelper.CurrentCaptureSize;](#)

———— Capture with camera(s) in the scene ————

- **Call the below method at any where you like:**

[ScreenshotHelper.iCaptureWithCamera\(camera, onCapturedCallback\);](#)

or

[ScreenshotHelper.iCaptureRenderTextureWithCamera\(camera, onCapturedCallback\);](#)

This method will automatically register([RegisterRenderCamera](#)) the camera to attach a [CameraOnRender](#) script on it.

If necessary, you can clear the texture and remove the script([CameraOnRender.cs](#)) by executing [iUnRegisterRenderCamera\(camera\)](#) or [iUnRegisterAllRenderCameras\(\)](#).

———— Get the currently stored texture ————

These parameters stored the image(Texture2D/Sprite/RenderTexture) since the last capture.

If you did not take any screenshot before, they will return a null value. Check null before access them.

[Texture2D texture2D = ScreenshotHelper.CurrentTexture;](#)

[Sprite sprite = ScreenshotHelper.CurrentSprite;](#)

[RenderTexture renderTexture = ScreenshotHelper.CurrentRenderTexture;](#)

Save Image (JPG/PNG)

- To save the captured image (within application folder):

```
new FilePathName().SaveTextureAs(texture2D, FilePathName.SaveFormat.JPG);
```

Set the SaveFormat enum for encoding the captured texture to different image format:

JPG - [FilePathName.SaveFormat.JPG](#)

PNG - [FilePathName.SaveFormat.PNG](#)

GIF - [FilePathName.SaveFormat.GIF](#) (Require Pro GIF)

Clear

When you have finished taking screenshots, you can call the Clear method to clean up memory:

```
ScreenshotHelper.iClear();
```

Thank you for downloading/buying this package!

Any problem and bug report please contact us at swan.ob2@gmail.com.

Remember to rate this asset on the AssetStore if you like it. Your review is always appreciated!

Download/Purchase on the Asset Store Now!!

Visit our asset page for more information.

<https://www.swanob2.com/assets>

SwanDEV