SWIFT SHADOWS

by Lost Polygon

Readme

v. 1.02

General information

Swift Shadows is a lightweight substitute for Blob Shadow Projector, heavily optimized for mobile devices. It works by doing a raycast to the surface and drawing a transparent quad that hovers above it. Literally hundreds of shadows can be drawn on a mobile device with a single draw call.

Features:

- Works on any platform, does not requires Pro license to work. Heavily optimized for mobile.
- Supports shadow texture atlases. All shadows in the scene can be drawn in a single draw call, even when they have different shapes.
- Tons of customization options.
- Shadows match object rotations.
- "Static" shadows that are calculated only once and produce no overhead in run-time. It is possible to have hundreds of static shadows without a performance hit even on mobile devices.
- Ease of use, absolutely no scripting required.
- Shadow generator tool for creating nice looking shadows automatically.
- *Blazing fast.* And I *really* mean it.

Limitations:

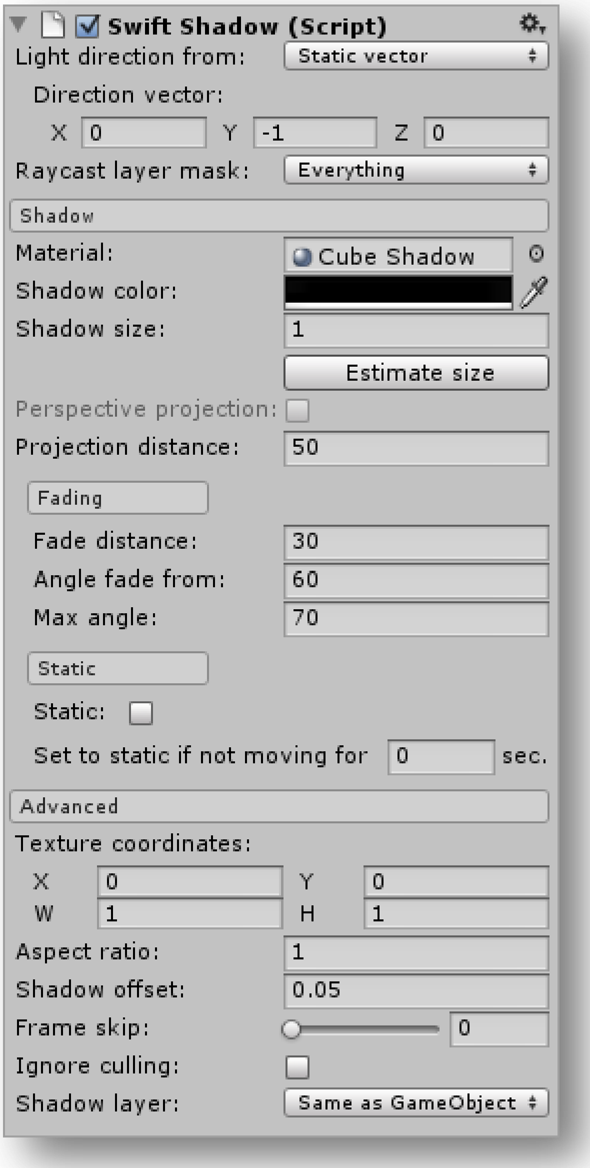
- Shadow textures are static. Shape of a single shadow is *not* updated in real-time. In some cases it is possible to work around this by using animated shadow cookie or by using multiple shadows.
- Each individual shadows only projects on one surface at a time. This leads to a discontinuity as shadow moves from one object to another.
- Shadow can extend the edges of object it is being projected on. It can also clip inside complex-shaped objects.

These peculiarities are rarely noticeable for moving objects with blurry edges.

Plugin is tested in Unity 3.5.7 - 4.3.4. Pro license is not required. Supported platforms: PC (Windows/Mac/Linux/Web), iOS, Android, Windows Phone 8, Windows Store, Flash.

Integration

No scripting required – just add *SwiftShadow* component to your object, hit Play and you'll see a fuzzy dark circle shadow right below the object.



☑ Swift Shadow (Script)

Light direction from: Static vector

Direction vector:

X 0 Y -1 Z 0

Raycast layer mask: Everything

Shadow

Material: Cube Shadow

Shadow color: ■

Shadow size: 1

Estimate size

Perspective projection: □

Projection distance: 50

Fading

Fade distance: 30

Angle fade from: 60

Max angle: 70

Static

Static: □

Set to static if not moving for 0 sec.

Advanced

Texture coordinates:

X 0 Y 0

W 1 H 1

Aspect ratio: 1

Shadow offset: 0.05

Frame skip: 0

Ignore culling: □

Shadow layer: Same as GameObject

Let's take a closer look on available options.

Light direction

Specifies the light direction vector. It can be entered manually, or calculated in run-time from a selected light source. You can pick any GameObject and it will act a point light source. Directional light sources also work as expected.

Raycast layer mask

Specifies layers on which the shadow will be projected. Usually, you must exclude the layer of the object itself from the layer mask, otherwise the shadow may collide inside of object and behave weird.

Material

Defines the material to be used for rendering the shadow. Shadows with the same material are automatically batched.

Shadow color

Defines the shadow color. Changing this does not break batching! The color data is written in the vertex color attribute.

Shadow size

Defines the size of the shadow.

Perspective projection

Shadow will grow and distort relative to the light source. This usually makes shadows look more realistic, but may result in artifacts at extreme angles.

Be aware that perspective shadows have lower performance that orthographic and are also harder to cull. Because of that, you may want not to enable this option when you have hundreds of shadows on mobile device.

This option can only be enabled if light source object is used.

Projection distance

Maximal distance from the object to the surface. Shadow will not appear beyond this distance.

Fade distance

Defines distance to the surface at which the shadow will start fading out. Shadow will totally fade out when reaching the Projection distance.

Max angle

Target won’t be drawn if shadow falls on the surface at an angle that is larger than this value.

Angle fade from

Defines the angle at which shadow will start fading out. Shadow will totally fade out when reaching Max angle.

Static

If you have shadows that are not moving throughout the gameplay, you can set them as static to get a massive performance boost. Static shadows are only calculated on the `Start()` and not calculated again automatically. Hundreds of static shadows can be drawn even on a mobile device with very little resource consumption.

Be aware, though, that static shadows are always drawn in a separate draw call to non-static shadows.

If you want to force an update for static shadows, just use this line of code:

```
    SS_ShadowManager.Instance.UpdateStaticShadows();
```

Set static if not moving for X sec.

Sometimes objects only move occasionally and stand totally still for rest of the time. This option conveniently allows you to make the object's shadow static automatically if the object is not moving for given time. The object will be automatically returned to non-static state if moved or rotated. This saves a lot of performance without the need of thinking whether the shadow should be manually set to static or not. Note, that this option has a small overhead, so setting Static manually is still the best option if you know the shadow won't ever change.

Texture coordinates

Defines the texture UV coordinates for the shadow. This is used if you have a shadow atlas to render multiple different shadows in a single draw call.

Aspect ratio

Defines the *width/height* aspect ratio of shadow frustum. Use this to get rectangular-shaped shadows.

Shadow offset

This value defines the distance at which the shadow hovers above the surface. It is usually recommended to keep this value as close to 0 as possible, but you may need to increase if you see shadow flickering with the ground (so-called Z-fighting).

Frame skip

Sometimes you don't need the shadow to be updated every frame. In that case, you can increase this value to get some performance boost. Value of 0 zero disables frame skip.

IGNORE CULLING

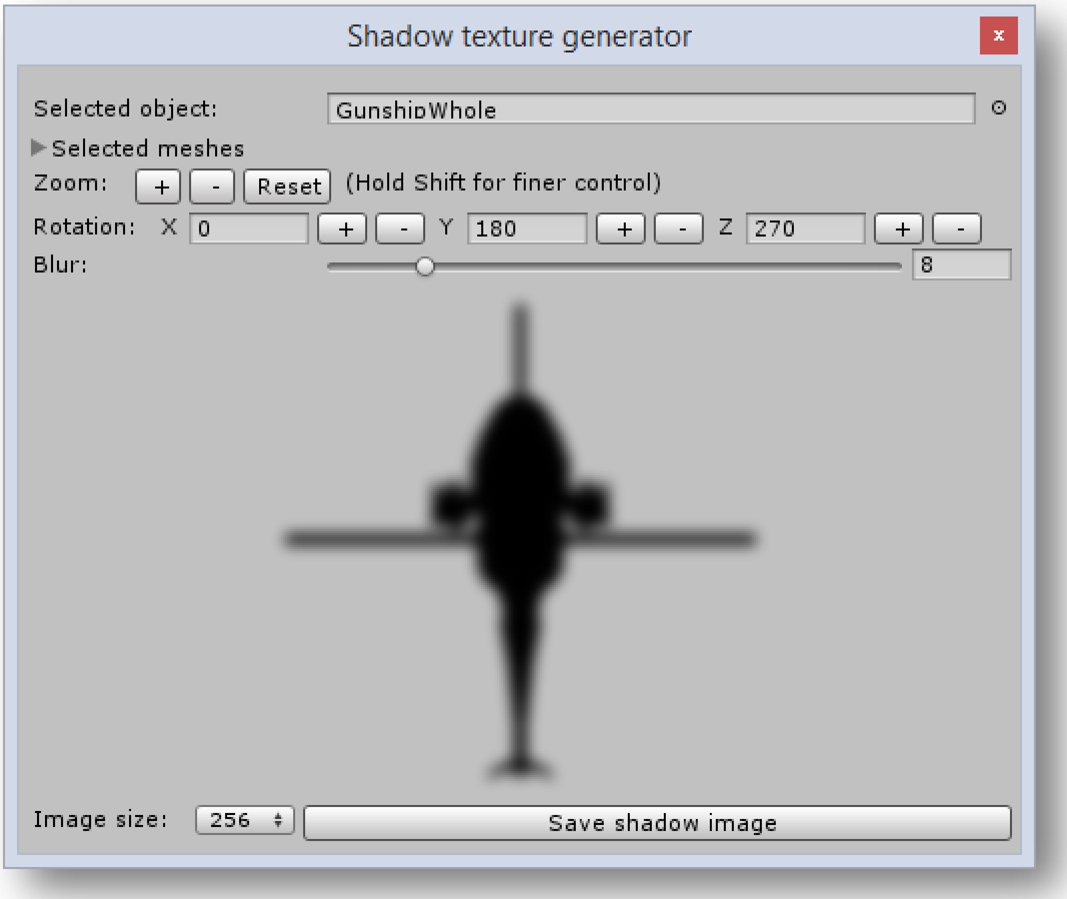
By default, the shadow is not calculated if it cannot be seen, to avoid wasting CPU resources. However, this check has some overhead by itself. Turn on this option if you know your shadow is always in camera's sight to get a small performance boost.

SHADOW LAYER

By default, shadow is drawn on the same layer as the GameObject. You can override this behavior and set the layer manually. This may help in certain situations.

Shadow texture generator

For objects with complex shape, a fuzzy circle shadow is not always the best choice. For that case, *Swift Shadows* includes a shadow texture generator that can save you a lot of time with trying to make a nice looking shadow texture manually.



Shadow texture generator

Selected object: GunshipWhole

Zoom: (Hold Shift for finer control)

Rotation: X 0 Y 180 Z 270

Blur: 8

Image size: 256

It is very simple to use: pick an object you want to generate the shadow for, select the object meshes you need to have in the shadow, rotate the object to "view" it from the above, adjust the blur amount to your taste, and save the resulting image! You can then use it with your custom shadow material to make shadows that exactly match the shape of your objects.

Let’s take a closer look on the generator interface.

Selected objects

GameObject to generate shadow for. You can select from prefabs or from current scene objects.

Selected meshes

Allows you to selectively disable meshes that you don’t need to appear on generated shadow texture. This is typically used for generating separate shadows for different parts of object. For example, for a helicopter you may want to generate separate images for the hull and for the blades, to allow animating them separately.

Zoom

Allows you to zoom on the center of the object. Holding Shift while pressing the button allows for more precise zooming.

Rotation

Allows you to rotate the object. In most cases, you’d want to produce a silhouette of an object as viewed from above. This is useful if your model has non-standard rotation.

Blur

The amount of blur to apply to a shadow texture. Value of 0 means no blur, value of 50 stands for maximum available blur.

Image size

The resolution of shadow texture to generate when saving. Texture will always be saved as a RGBA32 square texture with this size.

Troubleshooting

I see no shadows at all!

1. Make sure you are in Play mode. Shadows do not appear in Editor mode at the moment.
2. Check the **PROJECTION DISTANCE** value. Shadows will not appear beyond this distance.
3. Make sure the object you want shadow to project on has Collider attached. This is required for raycasting to work.

The shadow is twitching all around or appears in mid-air!

Make sure the RAYCAST LAYER MASK does not include the shadow caster itself, otherwise the shadow may cast on it as well. This usually happens when RAYCAST LAYER MASK is set to "Everything". It's usually best to put shadow casting objects on a separate layer to avoid this sort of problems too.

My shadows won't batch!

Shadows only batch if their materials and layers are the same. Also, static shadows are always rendered in a separate draw call.

In case of any troubles, enable Gizmos and select the problematic object. You should see the shadow frustum and the line of raycast. This may help you determine the source of problem.

Known bugs

- Shadows are only rendered in Play mode. This will be fixed in the update.

Contact

For any questions about this plugin, feel free to contact me at:
e-mail: contact@lostpolygon.com
Skype: serhij.yolkin

Changelog

1.0:

- initial release

1.01-1.02:

- minor fixes