Deckard Render

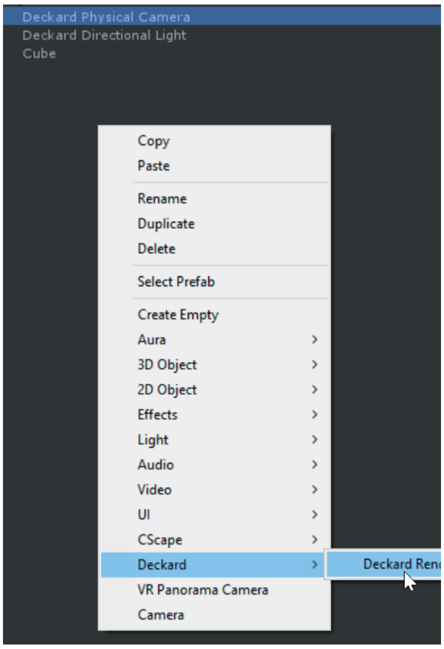
Documentation

Install

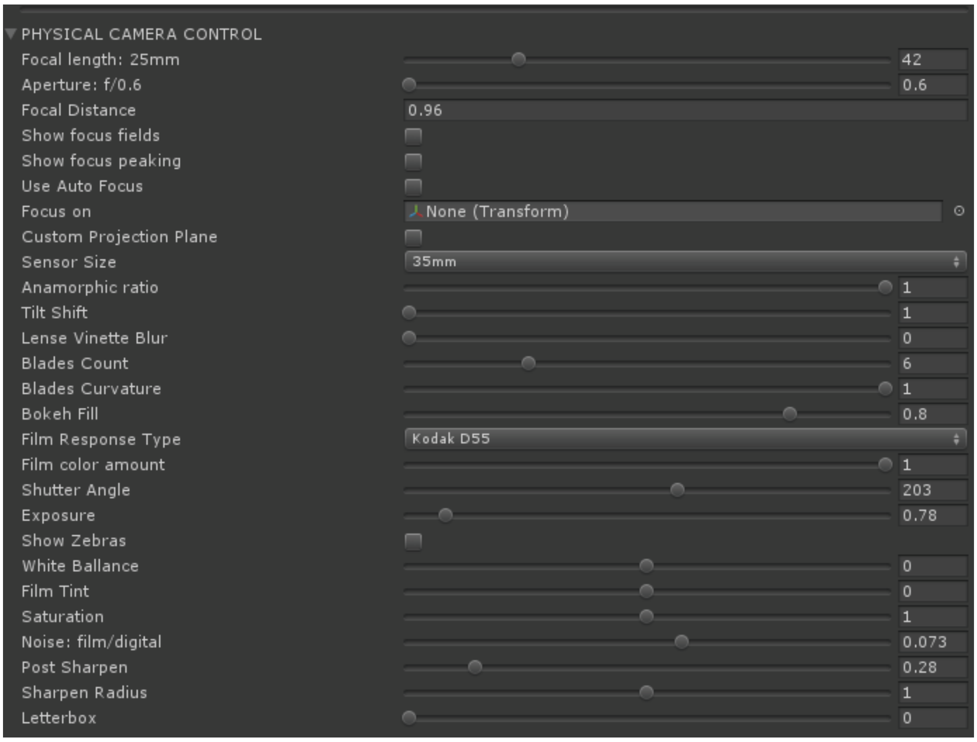
-Install it as any other Unity package. (Please don't change it's default directory location)
-Deckard Render uses Post processing Stack v1 due to it's better quality of Screen Space Reflections. For this reason install v1 instead of V2. You can find it here:
https://assetstore.unity.com/packages/essentials/post-processing-stack-83912

Usage:

-Add Deckard Render component to your camera
-To be able to see images rendered with Deckard, You have to open Deckard View. This can be done by going to Deckard Render/Deckard View in main menu in Unity. Or it can be done from Deckard Render component by Clicking a button Open Deckard View.
-Set it to your liking (almost all values have their tooltips that can be helpfull for understanding it's functions)
-if using any other image effects, please be sure to turn off any DOF and Motion Blur as those effects are managed by Deckard Render. If using Post processing Stack, place it right above Deckard Render script.
-Deckard Render can make any light behave as area light. You only have to set your light to realtime lighting, and add DeckardSoftLight component to it. Then set it's radius. For now, only quad area lights are supported.
-If you have moving objects in your scene and you want this object to cast a motion blur, you should add a Deckard Object Motion Blur component to it.
-Deckard render doesn't require Baked Lightmaps. It can work with them, but if you wan't dynamic lighting scenes, you can use only realtime GI baking.



Physical Camera Control



PHYSICAL CAMERA CONTROL

Focal length: 25mm	42
Aperture: f/0.6	0.6
Focal Distance	0.96
Show focus fields	□
Show focus peaking	□
Use Auto Focus	□
Focus on	None (Transform)
Custom Projection Plane	□
Sensor Size	35mm
Anamorphic ratio	1
Tilt Shift	1
Lense Vinette Blur	0
Blades Count	6
Blades Curvature	1
Bokeh Fill	0.8
Film Response Type	Kodak D55
Film color amount	1
Shutter Angle	203
Exposure	0.78
Show Zebras	□
White Ballance	0
Film Tint	0
Saturation	1
Noise: film/digital	0.073
Post Sharpen	0.28
Sharpen Radius	1
Letterbox	0

Deckard renderer uses a a Syntetic Camera Apreture tech - it samples multiple frames from different angles, and then merges them together - in this way it can manage multiple effects like soft shadows, depth of field, antialiasing and motion blur.
In this section you can set physical aspects of your Syntetic Camera:

-FOCAL LENGTH
this value defines a focal length value of real camera. While Unity uses vertical angle value for defining camera FOV (field of View). deckard translates this value into real world values for a sensor size/film frame size.

-APERTURE
While a real-world physical camera aperture defines both exposure and Depth Of Field (DOF), for a simplicity of usage, 'Aperture' value of Deckard Render influences only DOF, and not exposure.

-FOCUS DISTANCE
focus distance is a value that defines what parts of image should be in focus. It can be set to Auto Focus - and in this case it will put in focus objects that are in center of a screen. But it will work only on objects that are using Colliders. When selected, you can set a focusing speed.
Otherwise, if you set a target "FOCUS ON" object, it will automatically focus on that object. Numbers approaching 1 will be immediate focus, and smaller numbers slow focus.
Third, and default option is manual focus: Deckard will focus on a distance set on Focal Distance variable.
TIP: most movies use manual focusing - this is a best way to control your focus, and you should use it for better experience).

-SENSOR SIZE

Depth of a field is closely related to Focal Length and sensor size. Larger sensor size produces more blurring and les DOF. Here you can choose from various industry standard professional sensor/film sizes (IMAX, HASSELBLAD, 35mm and so on..). There are als some settings for non-professional sensor sizes like IPhone and Samsung S8. You can also set a custom size sensor.

-SHOW FOCUS FIELDS

This is a simple feature that can be used to easily set focus plane. It colorizes front and back plane with red and blue color. This has to be used only as a helper tool.
NOTE: this function works only when not using HDRP.

-**SHOW FOCUS PEAKING**

Instead of using Focus Fields, you can use focus peaking. Focus peaking is a found on most professional cameras and monitors. It works by detecting contrast, and it is a usefull tool to evaluate correct focus.

-ANAMORPHIC RATIO

Anamorphic format is the cinematography technique of shooting a widescreen picture on standard 35 mm film or other visual recording media with a non-widescreen native aspect ratio. It also refers to the projection format in which a distorted image is "stretched" by an anamorphic projection lens to recreate the original aspect ratio on the viewing screen. (It should not be confused with anamorphic widescreen, a different video encoding concept that uses similar principles but different means.)
This technique produces lense distortion, and stretches blur radius vertically. With this value you can simulate analog behaviour of such lenses.

-CUSTOM PROJECTION PLANE

Cuystom projection plane (OFF PLANE PROJECTION) is usually used in VR CAVE projections, or in a Projection Mapping. For this to work correctly, you have to create a Unity Plane object, deactivate it's mesh render and assign it into a Focus On field. Note: setting a custom projection plane will reset your camera, and Focal lenght/Focus Distance values won't have any influence anymore. Deckard will take FOV and focus from a plane. Also, for this to work correctly, be sure that you don't have activated any Antialiasing in post processing stack.

-FILM RESPONSE TYPE

This value instructs how a color should be threated. Deckard uses physical modelling of color response of various industry standard color responses. It will switch between analog modelliung of colors of various films from Fuji, Kodak, and digital camera color sciences from Canon, Blackmagic Cinema Camera, or standards like REC 709, or untreated colors. These color sciences are based on professional equipment, and (for now) doesn't simulate any color degradation effects. Some of them are to be used with future posprocessing in video color correction software.

-SHUTTER ANGLE

In photography, shutter speed or exposure time is the length of time when the film or digital sensor inside the camera is exposed to light, also when a camera's shutter is open when taking a photograph. This value influences motion blur effect. With shorter shutter speed there is less motion blur. With larger shutter speed there is more motion blur. Cinematic animation use 180 shutter angle speed as a most apropriate value for a smooth cinematographic motion.

-EXPOSURE

Here you can take controll of your image exposure. As DEckard Render works in HDR color space, we are avoiding any usage of ISO vs. Exposure. For simplicity of usage, aperture doesn't affect exposure, and there is no any ISO compensation for this. This means that you have much more flexibility when using deckard instead of a real camera.
NOTE: this function works only when not using HDRP. For exposure, please use Unity built in Post processing exposure.

-SHOW ZEBRAS

Zebras are usualy found in a professional cameras. They will show a burned zones of your image (over-exposed pixels). It is a good tool for evaluating exposure of your image.

-WHITE BALLANCE

Sometimes you want to make your image more cold or warm. Here you can set your white ballance value.

-FILM TINT

With this value you can colorize your image. It is usually used together with White Ballance to reoughly tune color look.

-SATURATION

With this value you can control a saturation of your image. This saturation happens before any film sensor processing.

-NOISE

As Deckard Render, unlike other raytraced renderers, has a completely noiseless output, this should be used to add a little bit of noise to your image. This noise is modelled after real sensor/film noise. Too much of a noise can be bad, and usually is avoided. But a little bit of noise/Grain can make your image look more organic and more detailed. A small anount of noise also adds to a perceived dynamic range of an image, helps video compression and helps in avoiding banding.
Values greater than 0 simulate digital sensors (ads noise) while negative values simulate film grain.

-POST SHARPEN/SHARPEN RADIUS

Post processing sharpen effect. Most TV content uses some kind of post processing sharpening effect. This effect can bring up some contrasts and make still images look better. But it also can introduce some aliasing issues. It can be used to simulate video degradation for creative purposes. POST SHARPEN is best when used for static image renderings.

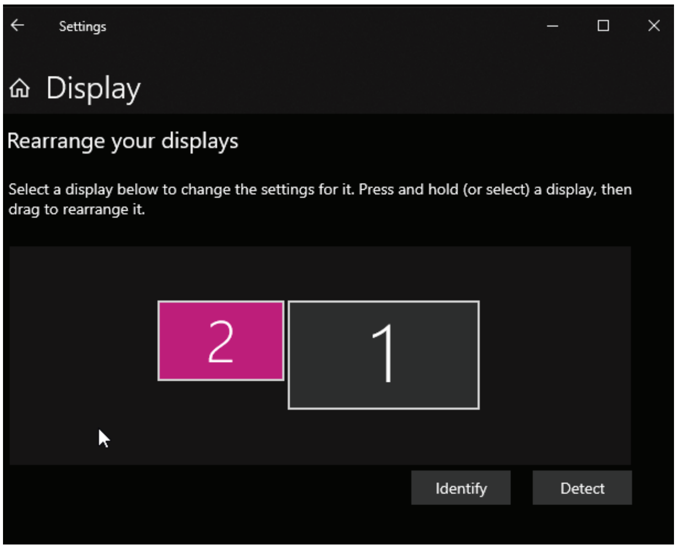
-LETTERBOX

Sometimes we want to use 16.9 image ffformat, but have an image that is letterboxed to another aspect ratio. This is a good feature to mask problems with some image effects like Screen Space Reflections.

Using Secondary Monitor for Deckard Previews

Deckard has a new possibility to use Secondary Desktop monitor for Live Preview. This is a pretty nice feature for those that need ability to preview renderings in a full screen while working in Unity.
At this moment, this is an experimental feature and it will work only on monitors/TV Set that support FullHD.

SETTING UP MULTIPLE MONITOR SETUP (Windows)



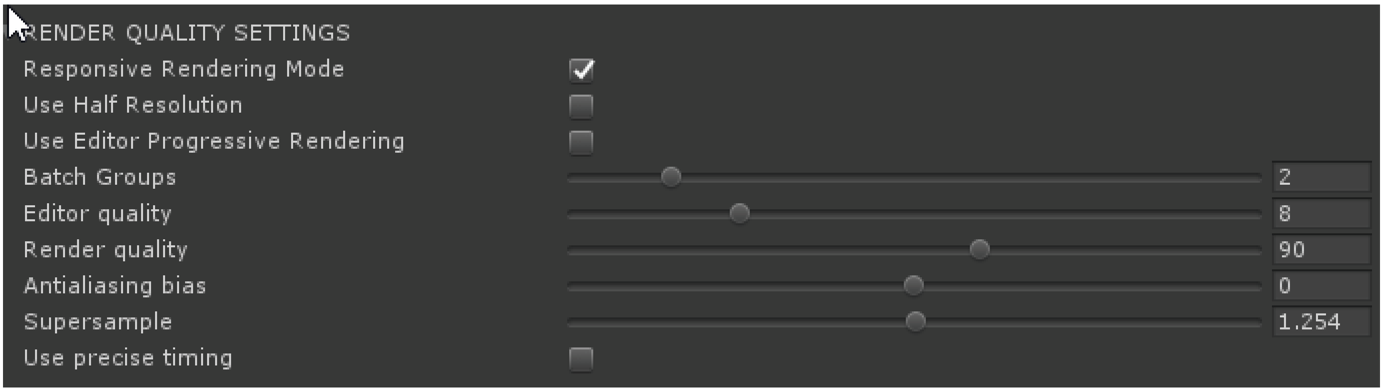
Rearrange your displays

Select a display below to change the settings for it. Press and hold (or select) a display, then drag to rearrange it.

-Attach your secondary FullHD monitor to your graphics card
-Open Windows Display Settings with a Right mouse click on Desktop. Go to “Multiple Dispays settings” and choose “Extend These Displays” .
-Set resolution for your secondary display to be 1920 x 1080
-Set your secondary display (Display 2) position to be on a Left from your main display (display 1)

In Unity go to Deckard Render/Deckard View and activate Deckard View window. This will activate Deckard.
In Unity go to Deckard Render/Secondary Monitor Display. Your preview image should show on a secondarly monitor in Full screen.

Render Quality Settings



RENDER QUALITY SETTINGS

Responsive Rendering Mode	☑
Use Half Resolution	☐
Use Editor Progressive Rendering	☐
Batch Groups	2
Editor quality	8
Render quality	90
Antialiasing bias	0
Supersample	1.254
Use precise timing	☐

Adjusting Render Quality Settings is essential for Unity Editor performance and final animation quality and rendering speed. This values should be set by keeping in mind your hardware and desired animation quality.

-RESPONSIVE RENDERING MODE (NEW)
As Deckard Rendering can be pretty CPU and GPU intensive, you will probably want to limit CPU and GPU time while working in Editor. This value sets Deckard to Responsive mode rendering, that will limit number of passes for eash editor Update - leading to faster Unity response, while slowing down Deckard Render previews.

-USE HALF RESOLUTION
This is a variable that can help your performance while working in editor. It should speed up rendering while working in editor by 4X. It lowers a final image resolution by 2X, resulting in 4 times less pixels to process. This value affects only In-Editor performance and not final animation quality or performance.

-USE EDITOR PROGRESSIVE RENDERING
This is a variable that can help your performance while working in editor. If selected, it will switch quality between Min and max progressive quality values, giving you a better performance when moving your camera or making changes in Deckard Render component.
NOTE: it isn’t advised to use this option while using RESPONSIVE rendering mode.

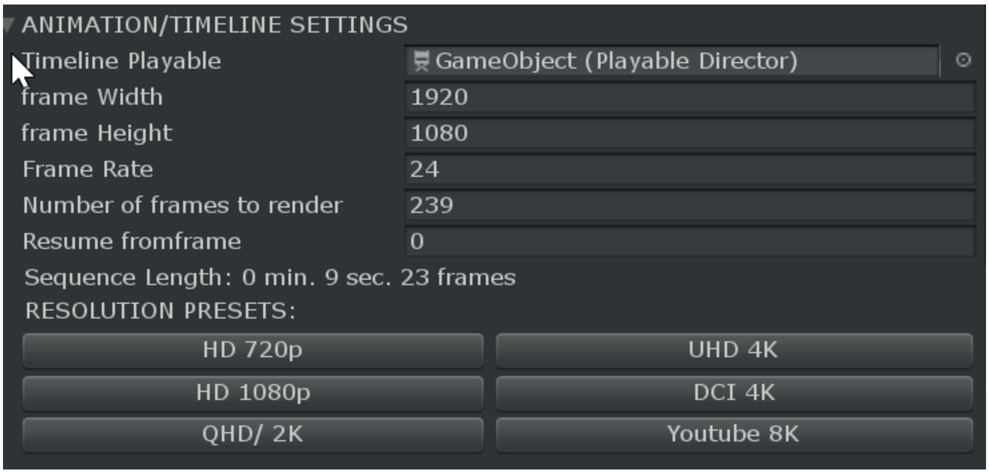
-EDITOR QUALITY - this defines how many Synthetic Aperture passes should be done on editor preview image. It should be as low as possible for maintaining FPS performance. while working in Unity editor.

-RENDER QUALITY - this defines how many Synthetic Aperture passes should be done while rendering final animation. More is better - but it really depends on factors as DOF and desired quality of your final image. Scenes with less DOF (more blurring) require less passes.

-ANTIALIASING BIAS - This value defines how much of multisample antialiasing should be applied on image sharp regions. Default value is 1

-SUPERSAMPLE - Supersamplig is a technique of rendering that uses higher resolution buffer for rendering into smaller image. It is one of the best techniques for dealing with aliasing. But it can be pretty demanding for graphics hardware. Bring it up only if Dekards default rendering doesn’t give you enough antialiasing.

Animation Settings



ANIMATION/TIMELINE SETTINGS

Timeline Playable: GameObject (Playable Director)

frame Width: 1920

frame Height: 1080

Frame Rate: 24

Number of frames to render: 239

Resume fromframe: 0

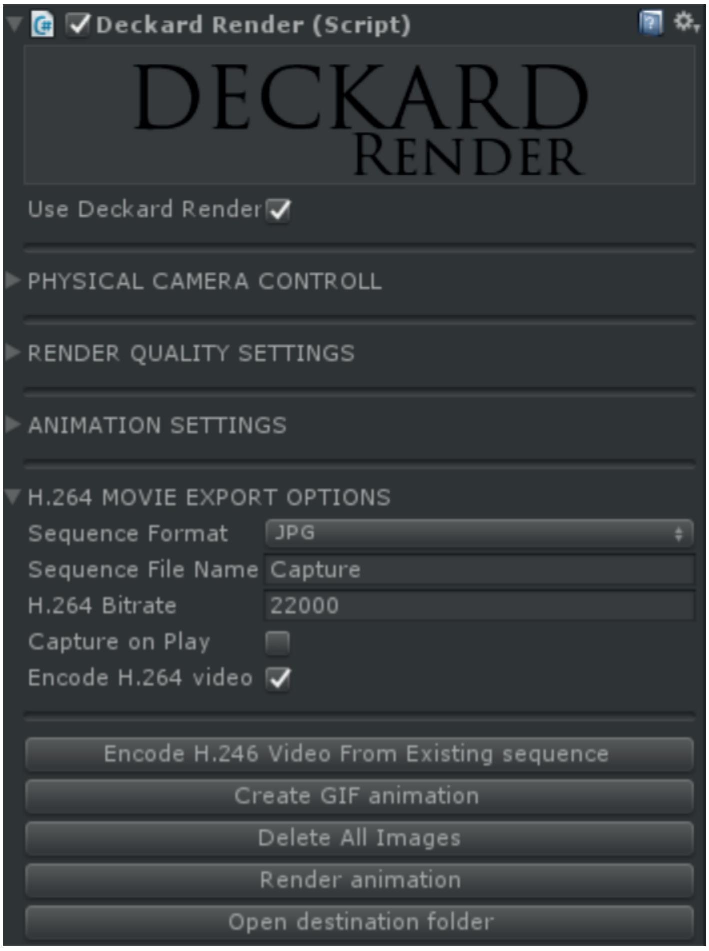
Sequence Length: 0 min. 9 sec. 23 frames

RESOLUTION PRESETS:

-Here you can set your final animation rendering settings - like resolution, framerate and animation range.
You can also find some buttons for setting your renderer to most used formats.

NOTE: If you are using Timeline for animation be sure to set here your timeline component. This will instruct Deckard on how to correctly calculate motion blur.

Animation Export Settings



Use Deckard Render ☑

▸ PHYSICAL CAMERA CONTROLL

▸ RENDER QUALITY SETTINGS

▸ ANIMATION SETTINGS

▾ H.264 MOVIE EXPORT OPTIONS

Sequence Format: JPG

Sequence File Name: Capture

H.264 Bitrate: 22000

Capture on Play ☐

Encode H.264 video ☑

Here you can set Video encoding options like bitrate and intermediary image format. JPG should be fine for most of use cases, while PNG or EXR formats are intended for image sequences that are intended to be color corrected in other editing applications.

IF YOU WANT TO EXPORT ANIMATION - BE SURE THAT CAPTURE ON PLAY CHECKBOX IS SELECTED

Using Real Motion Blur

Deckard Render can simulate real motion blur. It works by default on Deckard Render Camera and any objects that are animated by using Timeline.
If you are animating objects that move by scripted motion or physics, you have to add you have to add Dekard Motion Blur component on them.
Also, if you want to apply motion blur on Particles, you have to add DParticleMotionBlur script onto your particle systems.

NOTE: Particle motion blur can slow down your rendering time. - so use it only on particle systems that are short and move quickly (like explosions).

Due to closed API for Alembic importer, motion blur currently isn’t supported for Alembic animations.

Using Soft Area Lights

You can make any light behave as area light. Just add a Deckard Soft Light script to your light. (and be sure that your light is set to Realtime).
Then you can set it's size.
NOTE: Deckard smooths out any shadow acne, be sure to set your light Bias and Normal bias to min values.

TIP: When making outdoor scenes, you can add a Deckard Outdoor Ambient Light. It can behave as better AO tech giving you a nice soft shadow.,

Using Post Processing Effects

Deckard is capable of smoothing result of many post processing effects,. making them really shine. Due to it's internal working, it can make any image effect behave much better. There is one thing to know before applying any post processing effect:
-Always place any image effects between Deckard Pre Process and Deckard Render behaviours.

HOW TO CONFIGURE UNTIY POST PROCESSING STACK FOR DECKARD?
-For best appearance you should use Post processing Stack, but you should also turn off some of the effects that Deckard does better. Here is a list of most common scenario:

EFFECTS TO TURN ON:

-Antialiasing - Deckard uses it'ss own antialiasing methods, but also turning on Post processing antialiasing gives a smoother and better overall quality while keeping a possibility to use less rendering iterations.

-Bloom - Deckard doesn't take bloom into account so you should use the one from post processing stack. Note that you will have to tune it differently, but it will be much more stable and nicer.

-Screen Space Reflection - this one can be essential for a good looking render. My suggestion is to use Post Processing Stack v1 instead of V2, as it features more accurate reflections.

-Chromatic abberation - this one can give you a nice real lense effect that smoothes out colors.

-Grain - this one will be almost eliminated by deckard. But it gives a right feel of real camera. You can crank this value 70%.

EFFECTS TO TURN OFF:

-Depth of Field - this one should be turned off as it will clach with Physical modelling DOF integrated in Deckard Render.

-Motion blur - this one should be turned off as it will clach with Physical modelling motion blur integrated in Deckard Render. Resulting in accumulated motion blur that is incorrect.

-Dithering - it will crush your colors.

OPTIONAL:

-Vignette

-Eye adaptation - note that this one simulates automatic exposure - a feature that no-one would use in professional movie making. So use it only if you want a lo-fi look.

-Color Grading - I don't suggest to use it as there really isn't any need to do it when using Deckard. Because Deckard is physically modelled, it will give you nice and realistic colors from a start when using it's film modelling color science. You can eventually post process your videos in some Video Color correction application like Blackmagic DaVinci (free). Most of a fake imagery resides in wrong color correction.

Tips on Deckard Rendering

-Ambient occlusion is a nice tech for adding some realism. But you will notice that when working with Deckard, you actually won't need it. For outdoor scenes use Deckard Outdoor Ambient light, and for indoor scenes, use some nice area lights.
-As we arent using realtime rendering, turn off Lightmaps baking and use only Realtime Lighting. This will help you simplify your workflow.
-Deckard Area light are free, as they doesn't impact any performance. Be sure to add Deckard Area light to all lights in your scene as this script helps to deal with shadow pixelization and biasing issues.

Usefull Post Processing Effects -FREE

There are few image effects that can really give their best with Deckard, and all of them are free:

Light Shafts - performant volumetric lights:
https://github.com/robertcupisz/LightShafts

SEGI - realtime global illumination that works well with deckard
https://github.com/sonicether/SEGI

Screen Space multiple scattering - nice fog effect that takes into account light scattering
https://forum.unity.com/threads/screen-space-multiple-scattering.446647/

Using HDRP with Deckard

If you are planing to use Deckard with 2019.1 Unity version of HDRP you have to follow
this instructions:
-Important: Don't install DeckardPPS_HDRP_2018 package!! This is only to be used with Unity 2018 and PPS v2!!
-Add a Deckard component to your camera as usual.
-Deckard will prompt to Switch camera to Deferred HDR rendering. Instead click on "Don't switch, I'm using HDRP" button.
-In Unity go to Main Menu/Window/Deckard View. This will open a new preview window for Deckard and it will have a same resolution as your Game View, but there you can preview what Deckard renders.
-Use Deckard and have some fun.

Known Issues:
-Unity HDRP supports only one single directional light. Due to this limitation, you shouldn't use Deckard Ambient Outdoor light in
combination with another directional Light.
-Some combinations of Unity and HDRP can produce a memory leak when a directional light gets moved while in play mode. This bug affects Unity,
and having it with Deckard will eat all available memory much sooner - producing a system crash.
If you experience this issue, be sure not to use Dekcard Soft Light on diurectional lights, and be sure not to move Directional lights.Best way would be to
try different combinations of Unity and HDRP and see what version doesn't produce this bug.
-Unity HDRP raytrace works with Deckard, but with some limitaions - Raytracing requires much VRAM, and you might not be able to render
at larger resolutions with Deckard. 720p is known to work well on RTX2080 that has only 8GB of VRAM. RTX2080TI with 11gb of memory (or better) is recomended
when using raytrace.

p.s. At this moment some Deckard post processing features are unavailable with HDRP. You can compensate for those features by Using built in HDRP image effects.
p.s.s. Deckard Demo scenes doesn't work with HDRP.

Troubleshooting

Animated Character parts floating around

Some characters could have this bad habit of having problems with root objects, and this could trigger one of the bugs in unity. When this happens?

-your character uses multiple materials
-each part of a character uses different root bone objects

FIX:
-Select all skinned mesh parts (all character objects that use Skinned Mesh Renderer component)
-In Skinned Mesh Renderer script look for a value of Root Bone. You will notice that all objects probably have different Root Bones.
-In a hiearchy view find an object that is usually called "mixamo:Hips" (for Fuse characters). If you use your own custom characters, than this object chould be a root of your bones system.
-Drag and drop this object into a Root Bone field.
-Now, all your skinned mesh parts should have a same root bone object and your animation will play correctly.

Post Processing Stack effects not showing up in Unity 2019 (and later)

If you are using Built in PPS in Unity 2019.1 (or later) downloaded with Package Manager, and your postr processing stack effects aren't showing up, be sure to deselect "Directly To Camera Target" option in Post Process Layer script.

Deckard doesn't compile Video files on Mac OS?

It can happen that Deckard isn't able to compile your Video files from Image sequences. This can happen on some Mac OS systems when user doesn't have privilegies to execute FFmpeg.
If this happens, you have to call " chmod u+x" on the ffmpeg-binary (from terminal). This binary can be found in DeckardRender/StreamingAssets. Doing so will grant you privilegies to run external exacutables.