# VR Hands and FP Arms - Manual v1.2

## Introduction to VR Hands and FP Arms

Shaders need shader model 3.0+

PBR Arms and Hands for female and male characters. Assets use our custom and fast standard skin shader which supports features like: Translucency, Tint, Rim light, Ramp. With smart sliders you are able to control every aspect of the shader. Skin shader could be also used for other parts of character as basic skin shader. Models are animated and they contain pack of basic animations like: catching, throwing, fist fighting, idle, 3 spells, sword/knife fighting, pistol shooting.

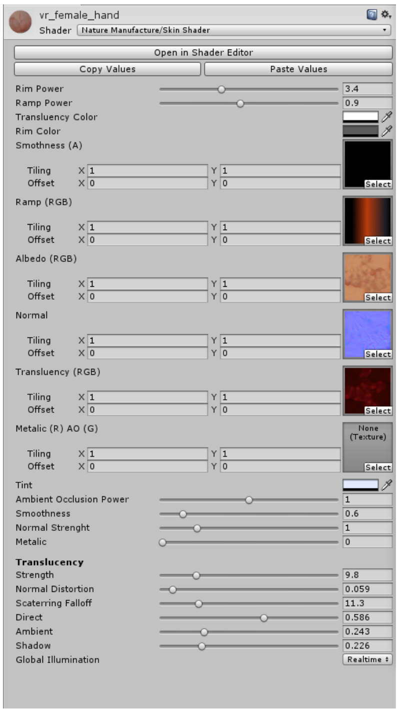
**Features:**

- Female and male arms for first person games
- Female and male hands for VR games
- Controllers for left and right hand/arms
- 16 Animations for Arms
- 21 Animations for Hands
- Standard PBR skin shader
- Test models: pistol and sword
- Example and test system to control your hands and arms
- Example and test VR and FP characters, prefabs
- Mecanim controllers for each hand
- 4096x4096 Arms and hands textures (3 nations variants)
- 256x16 Ramp texture
- blood and dirt masks

Support: Email contact: Naturemanufacture@gmail.com
Skype contact: dahrrrr Manual author: Bartłomiej Galas
Facebook: https://www.facebook.com/NatureManufacture-559454417506747/?fref=ts

# Table of Contents

# 1. Skin Shader



vr_female_hand
Shader: Nature Manufacture/Skin Shader

Rim Power 3.4
Ramp Power 0.9
Transluency Color
Rim Color
Smothness (A)
Tiling X 1 Y 1
Offset X 0 Y 0
Ramp (RGB)
Tiling X 1 Y 1
Offset X 0 Y 0
Albedo (RGB)
Tiling X 1 Y 1
Offset X 0 Y 0
Normal
Tiling X 1 Y 1
Offset X 0 Y 0
Transluency (RGB)
Tiling X 1 Y 1
Offset X 0 Y 0
Metalic (R) AO (G)
Tiling X 1 Y 1
Offset X 0 Y 0
Tint
Ambient Occlusion Power 1
Smoothness 0.6
Normal Strenght 1
Metalic 0

**Translucency**
Strength 9.8
Normal Distortion 0.059
Scaterring Falloff 11.3
Direct 0.586
Ambient 0.243
Shadow 0.226
Global Illumination Realtime

### Skin shader uses:

- Translucency: effect which is showing up when model is between light source and camera. You are able to control color and power of this feature by Translucency (RGB) and Translucency Color (slider).
- Rim Light: effect of the outline around the hand which is controlled by slider
- Ramp Color: effect of shadow mollification and colorization on model borders.

This shader could be used as skin for whole character, but you have to create proper translucency mask. By it you could create beautiful  humanoids or aliens.

**Skin Shader Dynamic** contains also mask (RGB), color, power for dynamic overlay like blood or dirt.

**Skin shader requires shader model 3.0!**

# 2. Mecanim Controller

Our Mecanim controllers are VR and FP friendly. You could test animations and controller by our script "**Hand Animator Manager**" or "**Hand Animator Manager VR** " for Unity 5.5+ .
Our example mecanim controllers use 3 values:

- Base state (integer)- animation type.  (**hold** "1","2","3",...."-" on your keyboard during playmode or scroll at HTC controller).
- Action (bool)- release animations (left mouse button or trigger in HTC controller).
- Hold  (bool)- few animations have hold functions. (right mouse button or grip at HTC controller).

For VR look at unity input system image **https://docs.unity3d.com/Manual/OpenVRControllers.html** .

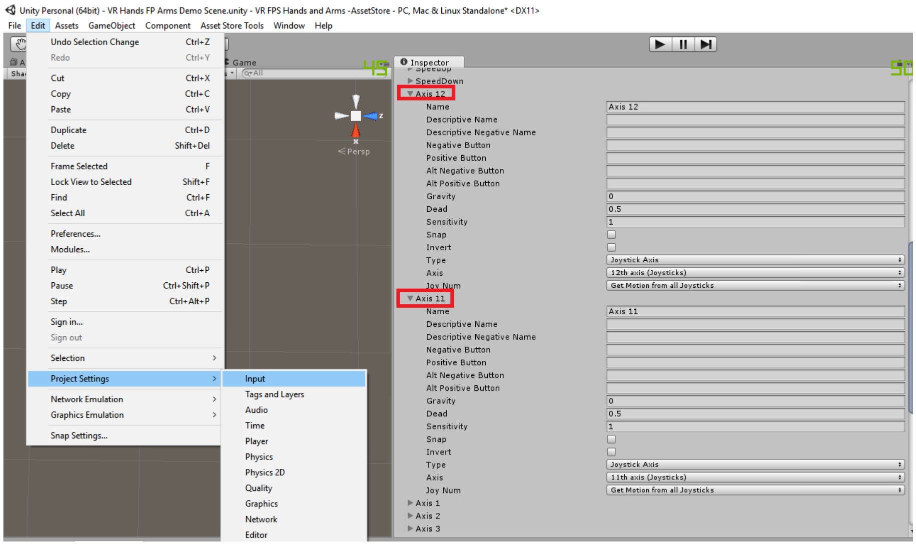
- Trigger - (7) which activate animations in our controllers.
- Grip - (8) which activate hold functions in our controllers.
- Scroll - (2) which change base state and animation types.

➢ For **FP Arms** Simply drag and drop our prefabs into scene and hit play, as we said you can control them by keyboard.

- For **VR hands** you could also test behave by keyboard like with FP Arms or test them with VR controllers (unity 5.5+) in 2 different ways:

**1. Way without SteamVR asset**

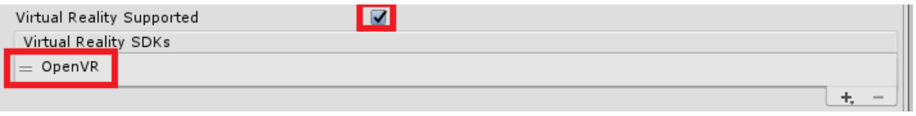
- Add 2 inputs to your project like in image below:



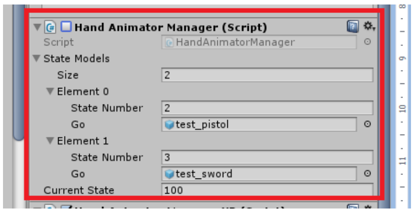
Axis 12
Name: Axis 12
Descriptive Name:
Descriptive Negative Name:
Negative Button:
Positive Button:
Alt Negative Button:
Alt Positive Button:
Gravity: 0
Dead: 0.5
Sensitivity: 1
Snap: □
Invert: □
Type: Joystick Axis
Axis: 12th axis (Joysticks)
Joy Num: Get Motion from all Joysticks

Axis 11
Name: Axis 11
Descriptive Name:
Descriptive Negative Name:
Negative Button:
Positive Button:
Alt Negative Button:
Alt Positive Button:
Gravity: 0
Dead: 0.5
Sensitivity: 1
Snap: □
Invert: □
Type: Joystick Axis
Axis: 11th axis (Joysticks)
Joy Num: Get Motion from all Joysticks

- Change this 2 values in player settings:


Virtual Reality Supported ☑
Virtual Reality SDKs
OpenVR

- Drag and drop our "VR_Character" prefab.
- **Disable Hand Animator Manager script on each hand prefab. It's used for keyboard test.**

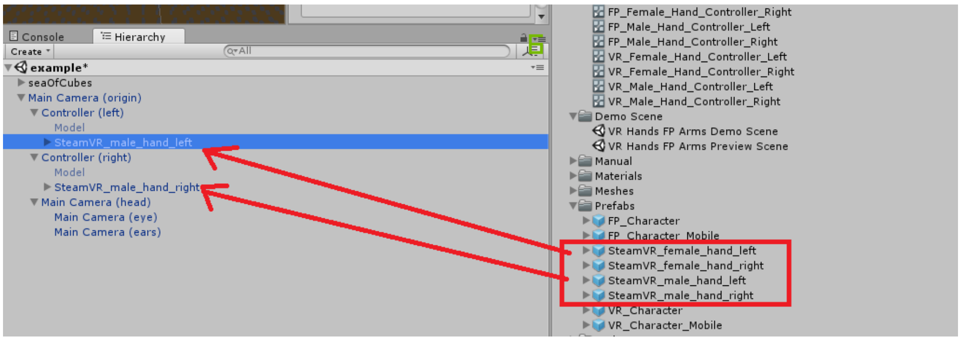

□ Hand Animator Manager (Script)
Script: HandAnimatorManager
State Models
Size: 2
Element 0
State Number: 2
Go: test_pistol
Element 1
State Number: 3
Go: test_sword
Current State: 100

- Test it - Hit play

Note: They will not follow your controllers!! You have to do it yourself or import SteamVR asset from asset store like in our second example.

**2. Way with SteamVR (similar for VRTK)**

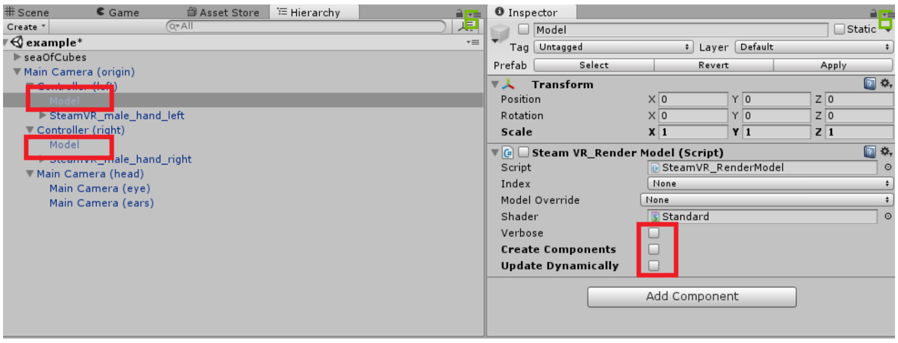
- Import SteamVR pack from assetstore if you don't have it already.
- Open SteamVR example scene
- Add our SteamVR prefabs as childs of SteamVR controllers - " Controller (left) and (right) like in image below:



example*
seaOfCubes
Main Camera (origin)
Controller (left)
Model
SteamVR_male_hand_left
Controller (right)
Model
SteamVR_male_hand_right
Main Camera (head)
Main Camera (eye)
Main Camera (ears)

FP_Female_Hand_Controller_Right
FP_Male_Hand_Controller_Left
FP_Male_Hand_Controller_Right
VR_Female_Hand_Controller_Left
VR_Female_Hand_Controller_Right
VR_Male_Hand_Controller_Left
VR_Male_Hand_Controller_Right
Demo Scene
VR Hands FP Arms Demo Scene
VR Hands FP Arms Preview Scene
Manual
Materials
Meshes
Prefabs
FP_Character
FP_Character_Mobile
SteamVR_female_hand_left
SteamVR_female_hand_right
SteamVR_male_hand_left
SteamVR_male_hand_right
VR_Character
VR_Character_Mobile

- Disable this values at Model objects in hierarchy to hide HTC controllers:



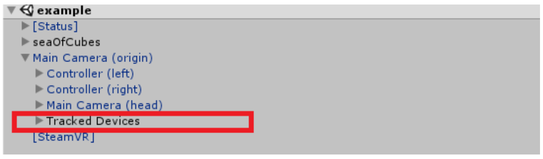
```
example*
  seaOfCubes
  Main Camera (origin)
    Controller (left)
      Model
      SteamVR_male_hand_left
    Controller (right)
      Model
      SteamVR_male_hand_right
    Main Camera (head)
      Main Camera (eye)
      Main Camera (ears)
```

□ Model □ Static
Tag: Untagged  Layer: Default
Prefab: Select | Revert | Apply

Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

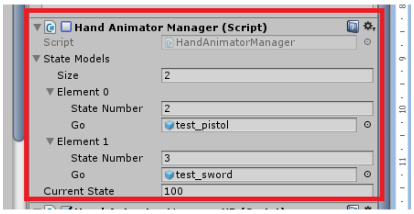
□ Steam VR_Render Model (Script)
Script: SteamVR_RenderModel
Index: None
Model Override: None
Shader: Standard
Verbose □
Create Components □
Update Dynamically □

- Remove "Tracked Devices" to hide HTC controllers:



```
example
  [Status]
  seaOfCubes
  Main Camera (origin)
    Controller (left)
    Controller (right)
    Main Camera (head)
    Tracked Devices
  [SteamVR]
```

- **Disable Hand Animator Manager script on each hand prefab. It's used for keyboard test.**



□ Hand Animator Manager (Script)
Script: HandAnimatorManager
State Models
  Size: 2
  Element 0
    State Number: 2
    Go: test_pistol
  Element 1
    State Number: 3
    Go: test_sword
Current State: 100

- Now hands will follow VR controllers and react on buttons! If they do not react, check if there is no errors about missing inputs message in your console. To check if all buttons react properly you could download this asset and test all inputs: https://www.assetstore.unity3d.com/en/#!/content/43621 This will take only 1 minute.

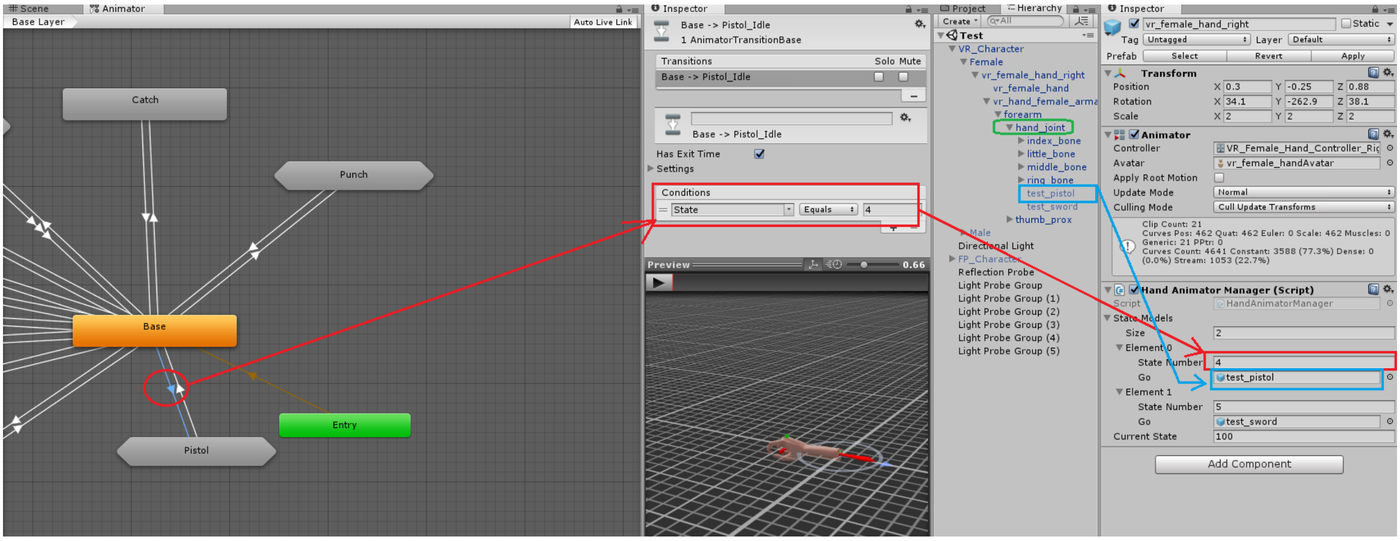
- Adjust offsets, rotations after first tests.

# 3. Weapons and items

Script "**Hand Animator Manager**" or "**Hand Animator Manager VR**"  are also used to test weapons and items. To catch or hold object you must use transform from "hand_joint" bone like in our example. Item could be parented in 2 ways:

- Child of the bone  like in our example.
- Outside object that follow "hand_joint"  bone.  Follow script object must be made your own.

Our script gives you ability to turn off/on objects, when proper state is actually used by controller. Look at image bellow.



Inspector

Base -> Pistol_Idle
1 AnimatorTransitionBase

| Transitions | Solo | Mute |
|---|---|---|
| Base -> Pistol_Idle | □ | □ |

Base -> Pistol_Idle

Has Exit Time ☑

Settings

Conditions

State | Equals | 4

Project | Hierarchy

Create | All

```
Test
  VR_Character
    Female
      vr_female_hand_right
        vr_female_hand
        vr_hand_female_arma
          forearm
            hand_joint
              index_bone
              little_bone
              middle_bone
              ring_bone
              test_pistol
              test_sword
            thumb_prox
    Male
  Directional Light
  FP_Character
  Reflection Probe
  Light Probe Group
  Light Probe Group (1)
  Light Probe Group (2)
  Light Probe Group (3)
  Light Probe Group (4)
  Light Probe Group (5)
```

Inspector

☑ vr_female_hand_right □ Static

Tag: Untagged  Layer: Default

Prefab: Select | Revert | Apply

**Transform**

|  | X | Y | Z |
|---|---|---|---|
| Position | 0.3 | -0.25 | 0.88 |
| Rotation | 34.1 | -262.9 | 38.1 |
| Scale | 2 | 2 | 2 |

☑ **Animator**

Controller: VR_Female_Hand_Controller_Ric
Avatar: vr_female_handAvatar
Apply Root Motion: □
Update Mode: Normal
Culling Mode: Cull Update Transforms

```
Clip Count: 21
Curves Pos: 462 Quat: 462 Euler: 0 Scale: 462 Muscles: 0
Generic: 21 PPtr: 0
Curves Count: 4641 Constant: 3588 (77.3%) Dense: 0
(0.0%) Stream: 1053 (22.7%)
```

☑ **Hand Animator Manager (Script)**

Script: HandAnimatorManager
State Models
Size: 2
Element 0
State Number: 4
Go: test_pistol
Element 1
State Number: 5
Go: test_sword
Current State: 100

Add Component

# 4. Upcoming Features

- Additional hand models ( knight, mage, soldier, etc..),
- More animations and  bigger example controller,
- More weapons support,
- wet hands surface support
- variants textures fixes

Support:    Email contact:    Naturemanufacture@gmail.com
Skype contact:   dahrrrr   Manual author: Bartłomiej Galas
Facebook:  https://www.facebook.com/NatureManufacture-559454417506747/?fref=ts