

# Bake Camera from Alembic

```
import hou

obj = hou.node("/obj")

ocam = hou.node(hou.ui.selectNode(node_type_filter=houd.nodeTypeFilter.ObjCamera))

reslist = ["1920x1080", "3840x2160", "3072x2109", "2224x1548", "Keep Original", "Custom"]
sl = hou.ui.selectFromList(reslist, message="choose resolution", sort=True, exclusive=True)
stopval = sl[0]
resolution = reslist[stopval]
if resolution is "Keep Original":
    pass
elif resolution is "Custom":
    custom = hou.ui.readInput("Enter resolution as #x# format, ie: 1920x1080. The x is needed.", title="Enter Resolution")
    resolution = custom[1]
    ocam.parm("resx").set(resolution.split("x")[0])
    ocam.parm("resy").set(resolution.split("x")[1])

else:
    ocam.parm("resx").set(resolution.split("x")[0])
    ocam.parm("resy").set(resolution.split("x")[1])

try:
    shotinfo = hou.node("/obj/SHOTINFO")
    backplate = shotinfo.parm("backplate")
except:
    backplate = " "
    pass

camName = ocam.name() + "_baked"
#setback = ocam.parm("vm_background").set(backplate)
tcam = obj.createNode("cam", camName)
tcam.moveToGoodPosition()
```

```
#copy keyframes
cam_xform=["tx", "ty", "tz", "rx", "ry", "rz"]
cam_parms=["resx", "resy", "aspect", "focal", "aperture", "orthowidth", "shutter", "focus", "fstop"]
tcam.parm("vm_background").set(backplate)

parms_bake = list()
parms_bake.extend(cam_xform)
parms_bake.extend(cam_parms)
start = hou.playbar.playbackRange()[0]
end = hou.playbar.playbackRange()[1]

with hou.undos.group("bake cam"):
    for x in range(int(start), int(end + 1)):
        hou.setFrame(x)
        tcam.setWorldTransform(ocam.worldTransform())
        for p in parms_bake:
            parm = tcam.parm(p)
            if parm.name() in cam_xform:
                parm.setKeyframe(hou.Keyframe(parm.eval()))
            else:
                parm.setKeyframe(hou.Keyframe(ocam.parm(p).eval()))
```

---

Revision #4

Created 25 April 2023 17:13:18 by Anthony

Updated 30 January 2024 17:37:22 by Anthony