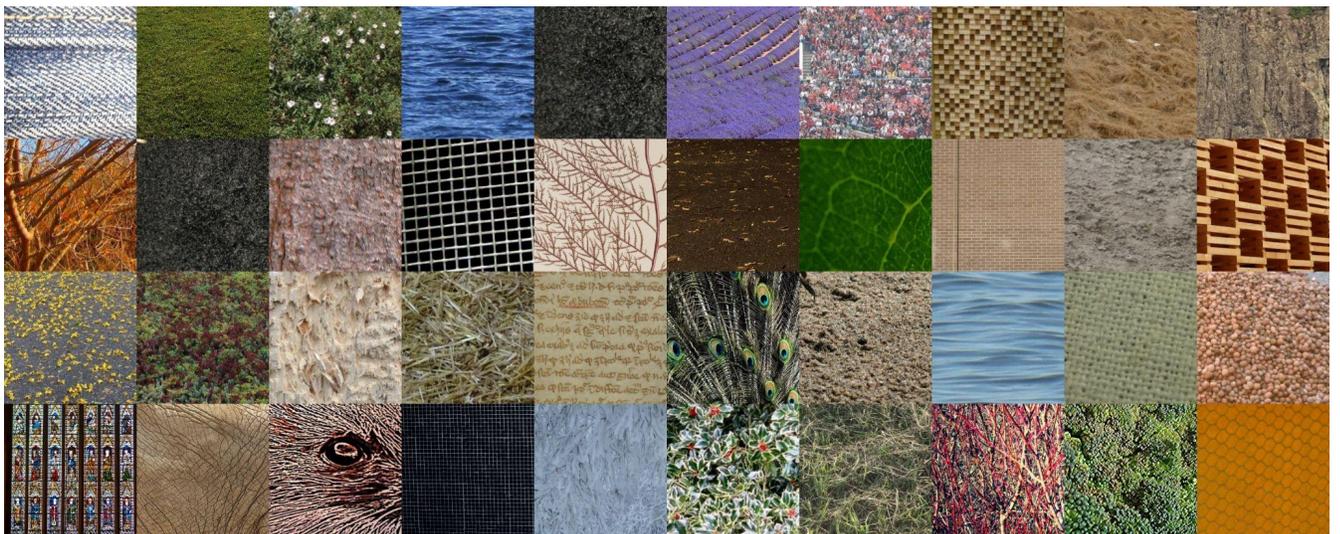
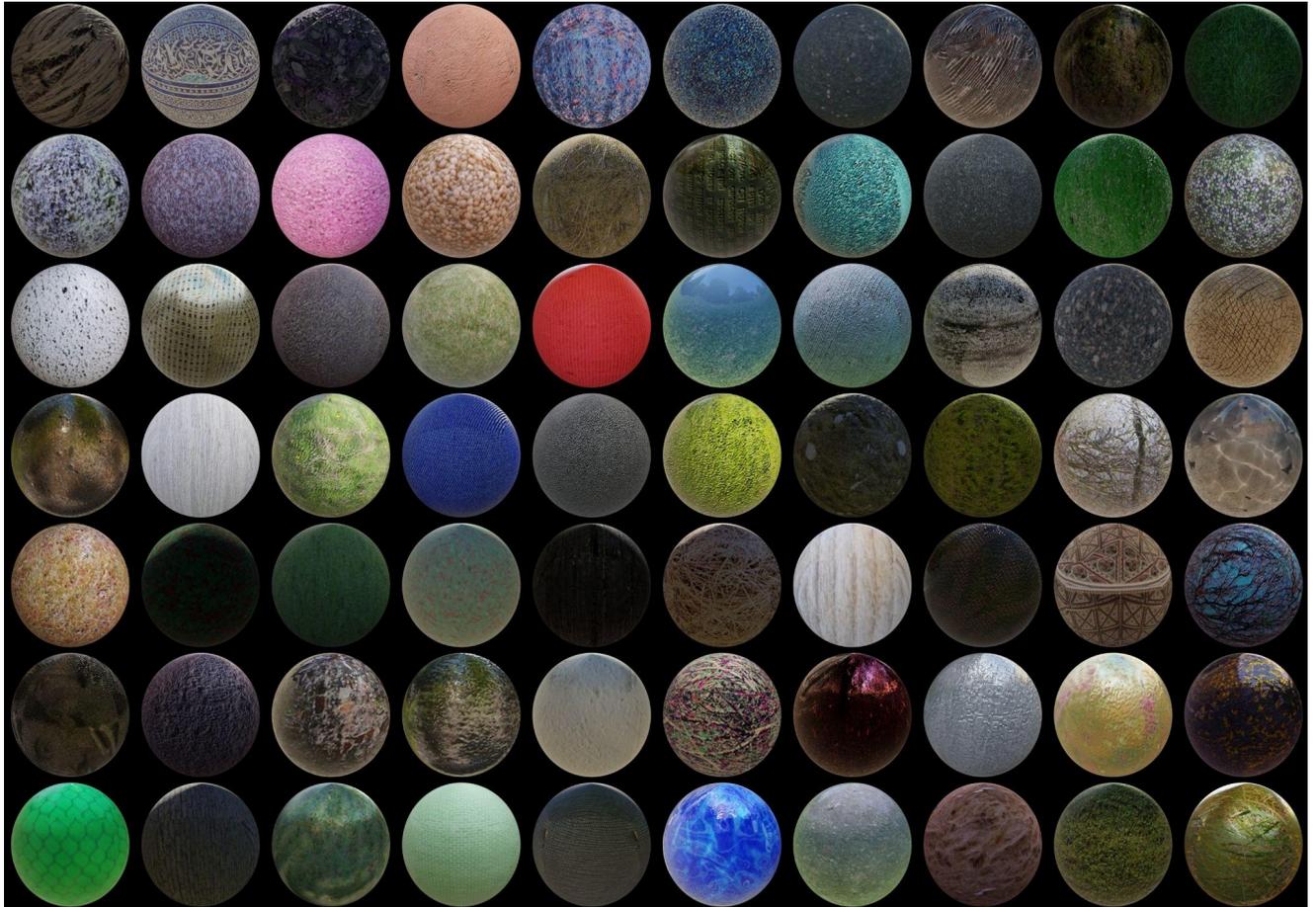
# Generation Code for infinitexture: Unsupervised extraction of textures and PBR materials from images

# Content:

1.  **Extract_Textures.py:** Receives a folder of random images and extracts regions of uniform textures, crop and save to file.
2.  **Turn_Texture_To_PBR.py:** Turn images of uniform textures (1) into PBR materials.
3.  **Mix_PBRS.py:** Mix existing PBRS materials(2) to generate more diverse PBRs.

## Extracting textures from images:

Identify regions with uniform textures in images, crop and save the textures as new images.

Script: **Extract_Textures.py:**

**How to use:**
Set input folder with random images to **Image_dir**
Set output folders where the texture will be saved to **out_dir**
The script should run out of the box with the sample data.

## Filtering smooth textures

Many of the textures extracted using **Extract_Textures.py** are simply uniform colors or black or white regions, and for most applications should be removed.

Script: **Remove_Uniform_Regions.py** Identify and remove textures of uniform colors.

**How to use:**
Set input texture folder containing textures to be filtered to **Image_dir**
Set output folder where the textures that were removed will be saved to **out_dir**
The script should run out of the box with the sample data.

## Generation SVBRDF/PBR materials map from images

Generate PBR materials from a texture by using the various properties of the RGB texture to generate maps for various PBR properties (Roughness, normals, metallic).

Script: **Turn_Texture_To_PBR.py**

**How to use:**
Set input texture image folder into **texture_dir**
Set output folder where the textures that were removed will be saved into **pbr_dir**
The script should run out of the box with the sample data.

# Mixing SVBRDF/PBR materials

Another way to generate PBRs is by mixing existing PBRS to generate new ones.

Script: **Mix_PBRS.py**

**How to use:**
Set the folder that contains PBR materials (each material as a subfolder) into **pbr_dir**
Set the output folder where the new mixed PBR will be saved in **merge_dir**
The script should run out of the box with the sample data.

**Note** this script assumes that the PBR materials file names follow that of what is generated by **Turn_Texture_To_PBR.py,** if you use different types of PBRs you might need to change the file names.

## How this works:

a) Divide the image into a grid. For every grid cell extract distribution of colors and gradients. Identify a region for which
all the cells have similar distributions as a uniform texture. Pick random channels from the extracted
texture image, augment them, and use the resulting maps as property maps (roughness, metallic,
height...) for the SVBRDF/PBR material.