Red Pill

AI RENDERING

Ushio

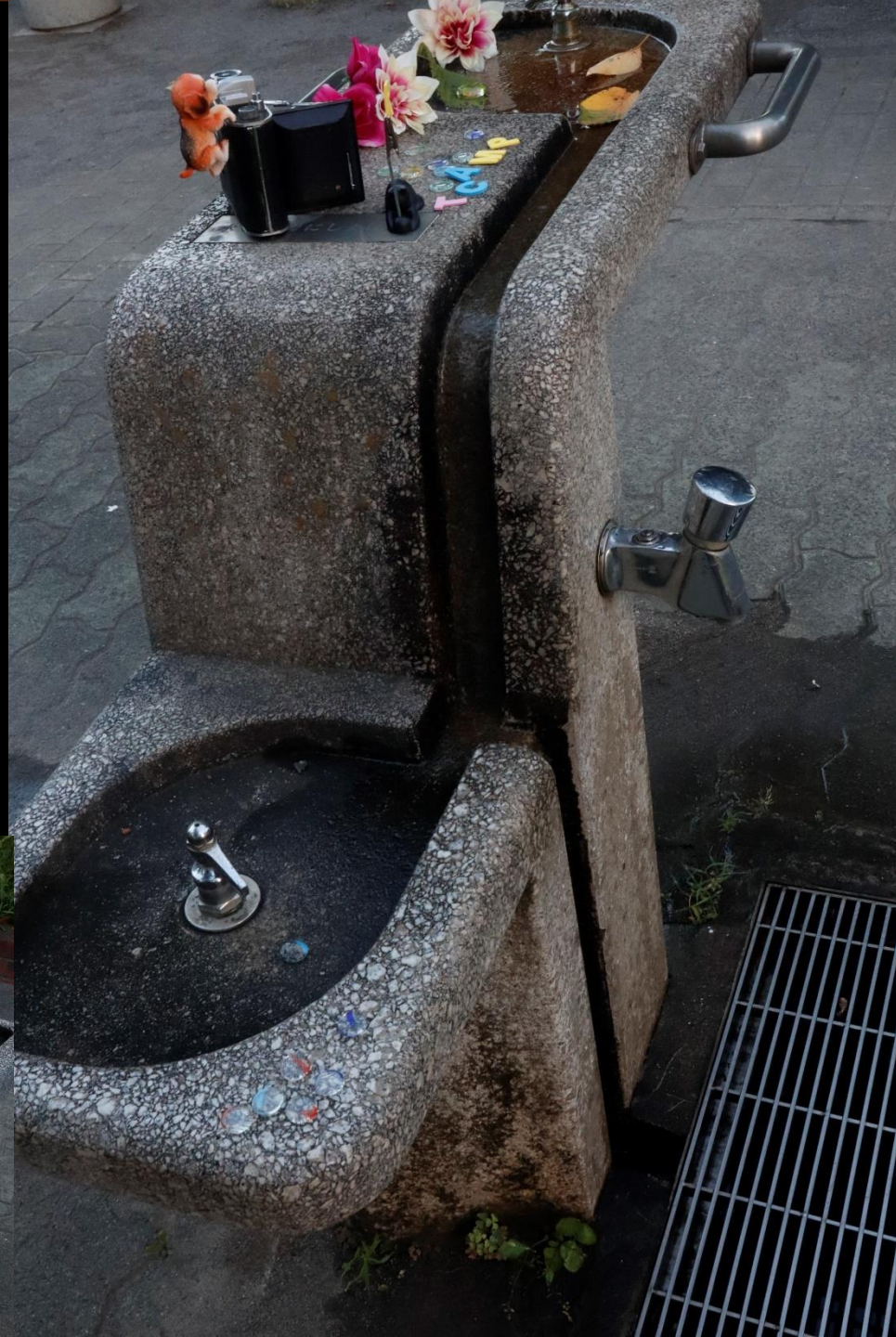
Implementation

- NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis
- Instant Neural Graphics Primitives with a Multiresolution Hash Encoding

Inputs

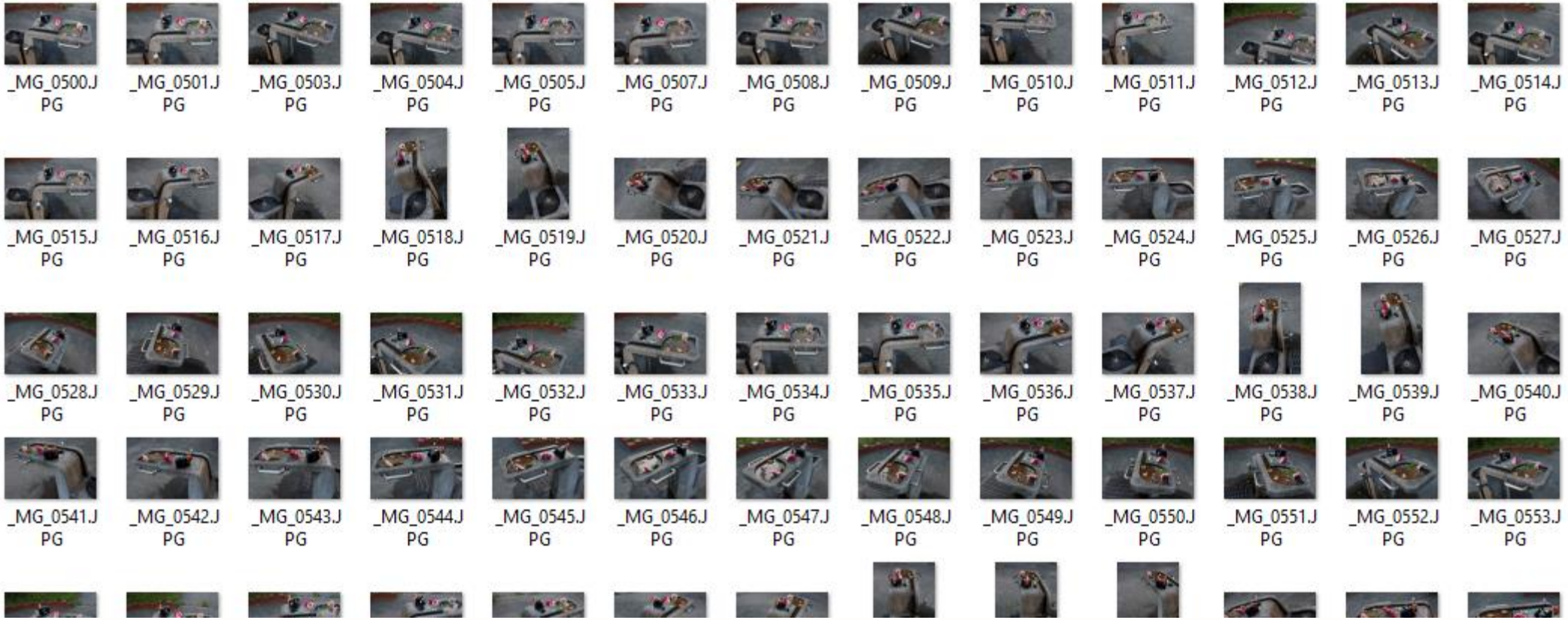

R
TCAMP


CAMR

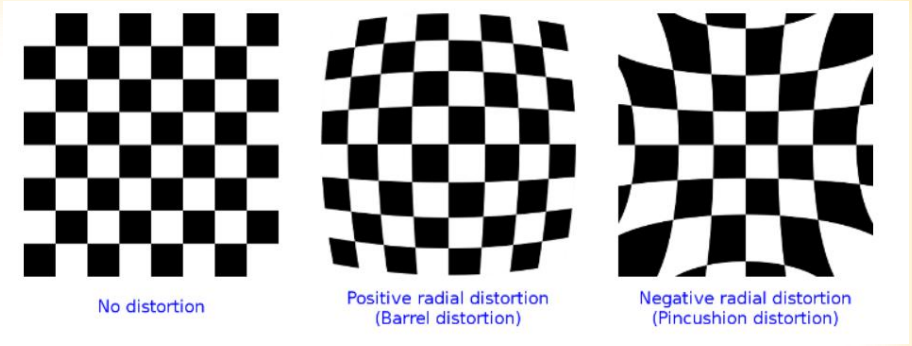


127 images – 3000 x 2000

Canon EOS M200



OpenCV Camera Model



- Real-world cameras have lens distortion
  - It is described as “Intrinsic parameters” on OpenCV Camera Model
  - Intrinsic parameters are calculated with camera pose on COLMAP (https://colmap.github.io/)

$$s\, m' = A[R|t]M'$$

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + t$$

$$x' = x/z$$

$$y' = y/z$$

$$x'' = x' \frac{1+k_1r^2+k_2r^4+k_3r^6}{1+k_4r^2+k_5r^4+k_6r^6} + 2p_1x'y' + p_2(r^2+2x'^2)$$

$$y'' = y' \frac{1+k_1r^2+k_2r^4+k_3r^6}{1+k_4r^2+k_5r^4+k_6r^6} + p_1(r^2+2y'^2) + 2p_2x'y'$$

where $r^2 = x'^2 + y'^2$

$$u = f_x * x'' + c_x$$

$$v = f_y * y'' + c_y$$

GPU Specific

- Thread-group matrix multiplications
- WMMA (NV hardware)

Thank you