IBGS: Image-Based Gaussian Splatting

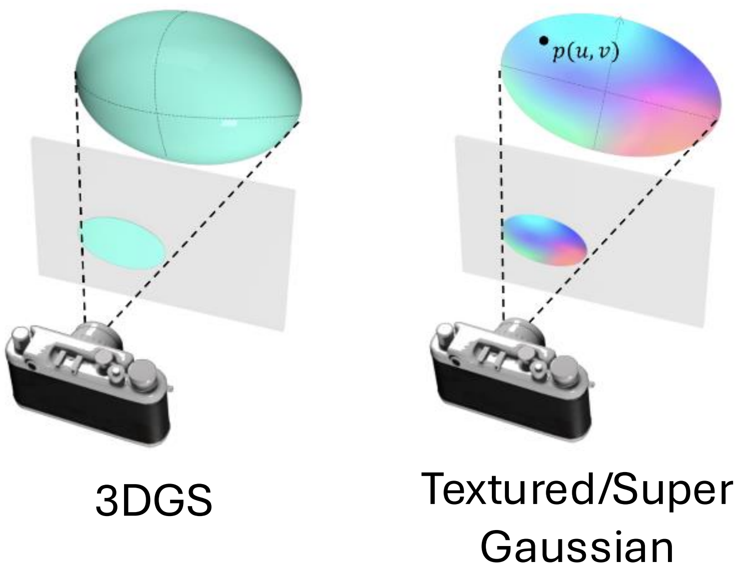
Hoang Chuong Nguyen[1], Wei Mao, Jose M. Alvarez[2], Miaomiao Liu[1]

[1]Australian National University, [2]NVIDIA

NeurIPS 2025

Motivation and Contribution

- 3DGS: model Gaussian color using a low-degree spherical harmonic
  - Handle complex view-dependent color.
  - Render images with high-frequency details.
- TexturedGaussian/SuperGaussian: learns spatially varying color of each Gaussian.
  - Render images with more fine-grained details.
  - Still cannot handle complex view-specific color.
  - Increase the storage footprint significantly.

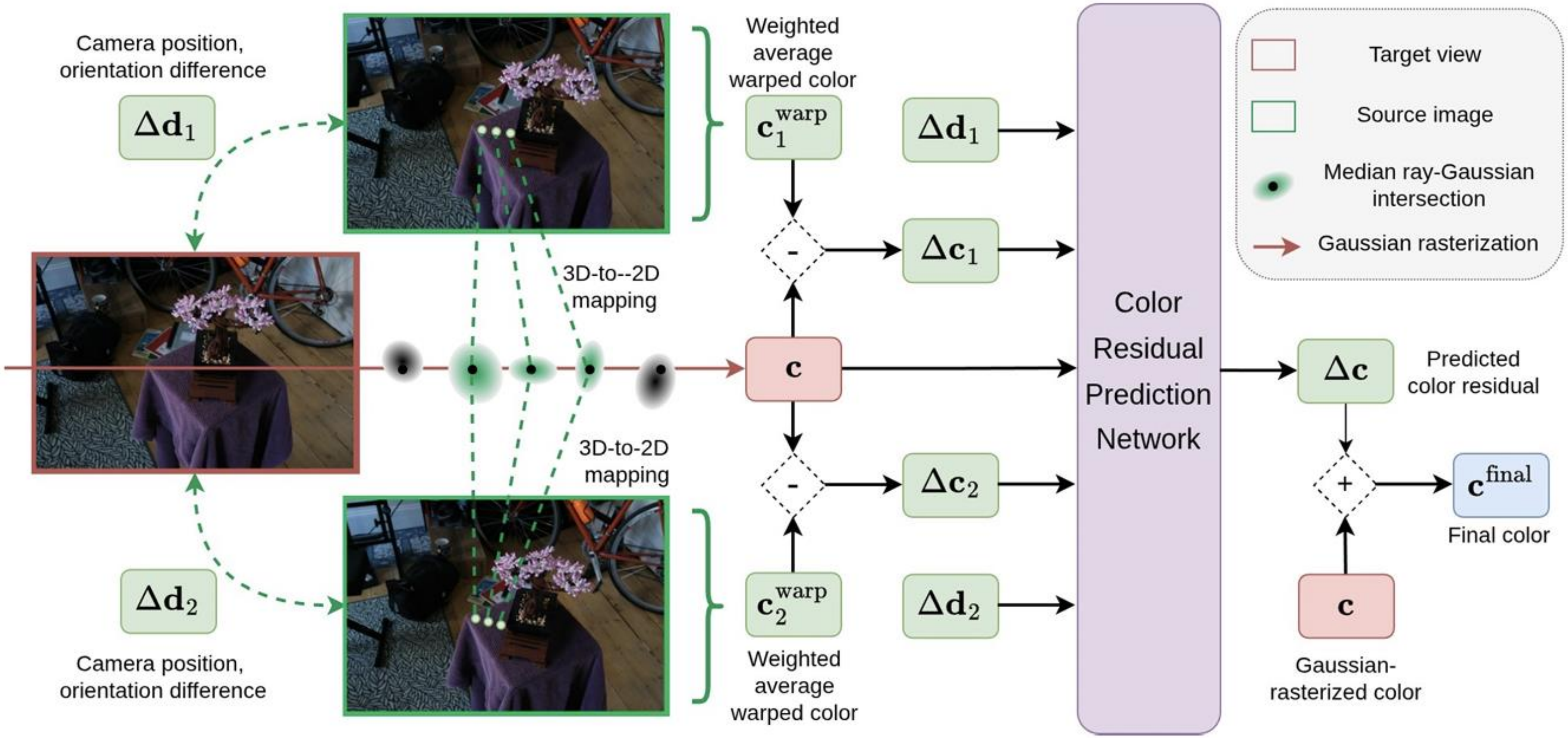


(figures obtained from SuperGaussian)

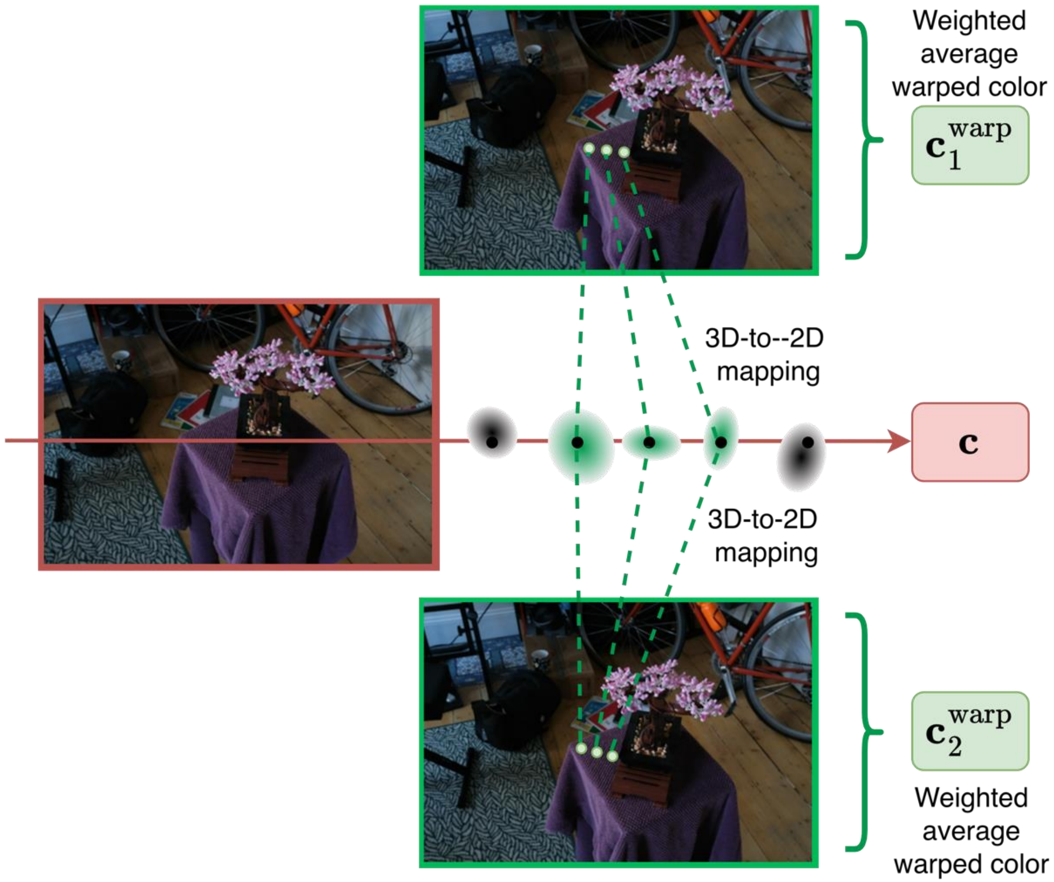
- Image-based Gaussian Splatting (IBGS):
  - Render images with fine-grained details and accurate view-specific color
  - No storage overhead.
  - Correct the exposure of an image rendered at any viewpoint.

Method overview

$$\mathbf{c}^{\text{final}}(\mathbf{p}) = \underbrace{\sum_{i=1}^{N} w_i\,\Psi_l(\mathbf{h}_i, \mathbf{v}_i)}_{\text{Base color } \mathbf{c}(\mathbf{p})} + \underbrace{\mathcal{F}\Big(\mathbf{c}(\mathbf{p}),\, \mathbf{d}(\mathbf{p}),\, \{\Delta\mathbf{c}_m(\mathbf{p})\}_{m=1}^{M},\, \{\Delta\mathbf{d}_m(\mathbf{p})\}_{m=1}^{M}\Big)}_{\text{Residual } \Delta\mathbf{c}(\mathbf{p}) \text{ predicted from multi-view observations}},$$



Color Residual Prediction Module

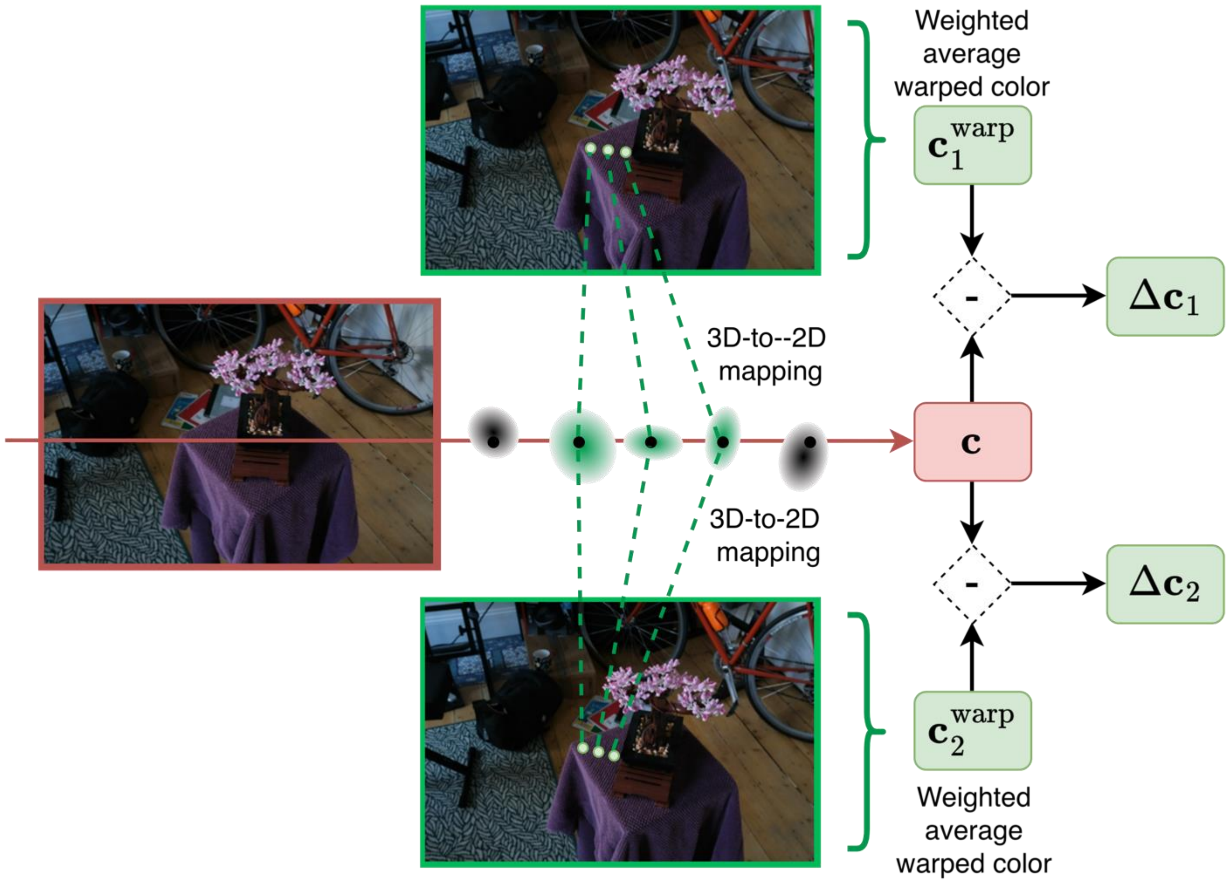


1. Base color **c** can is obtained via standard Gaussian rasterization.

2. Computing the warped colors:
   - Project camera-ray Gaussians onto multiple training images.
   - Obtain the warped colors at the projected pixels.
   - Compute the weighted average warped colors.

⇒ High-frequency details and view-dependent color information are captured in the multi-view warped colors $\mathbf{c}_m^{\mathrm{warp}}$.

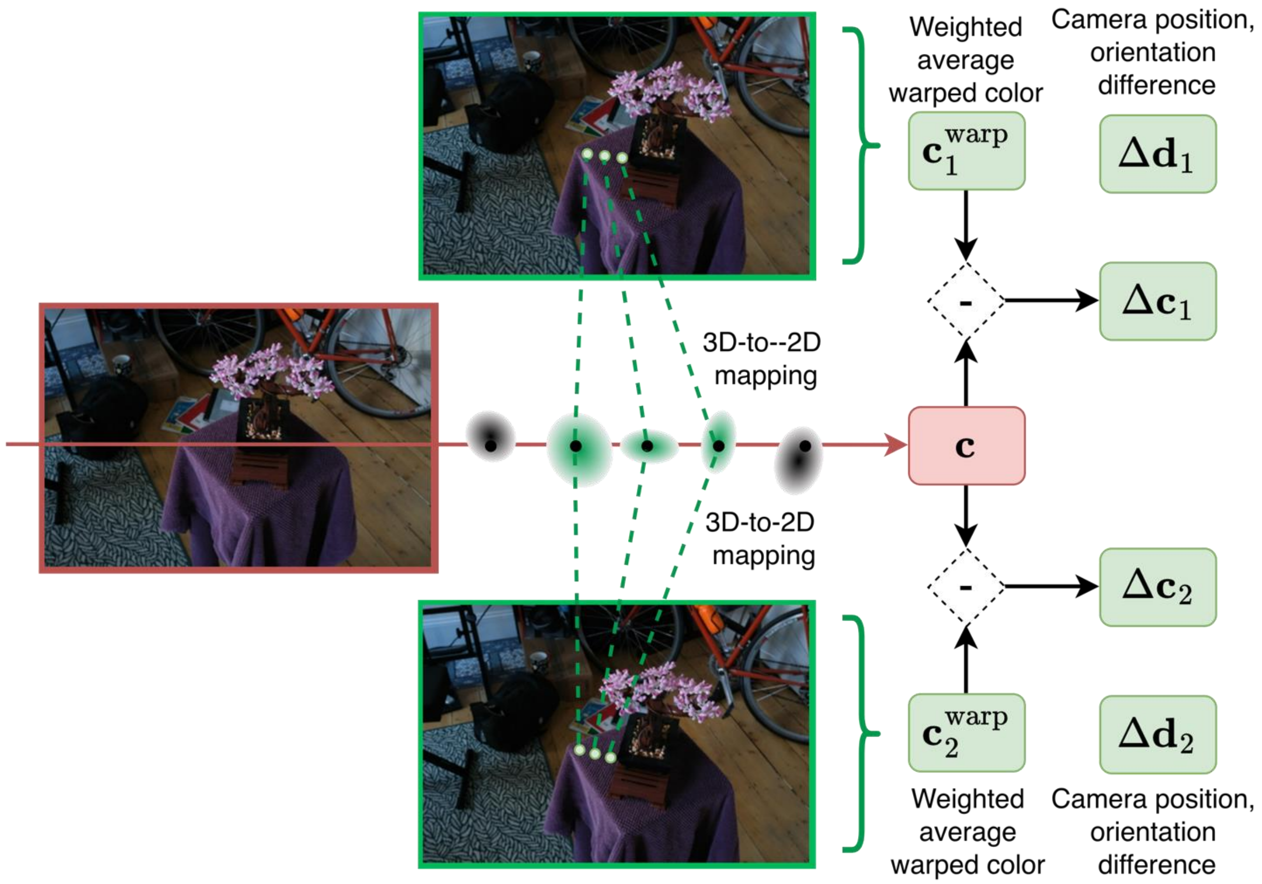
Color Residual Prediction Module



3. Compute the color difference:

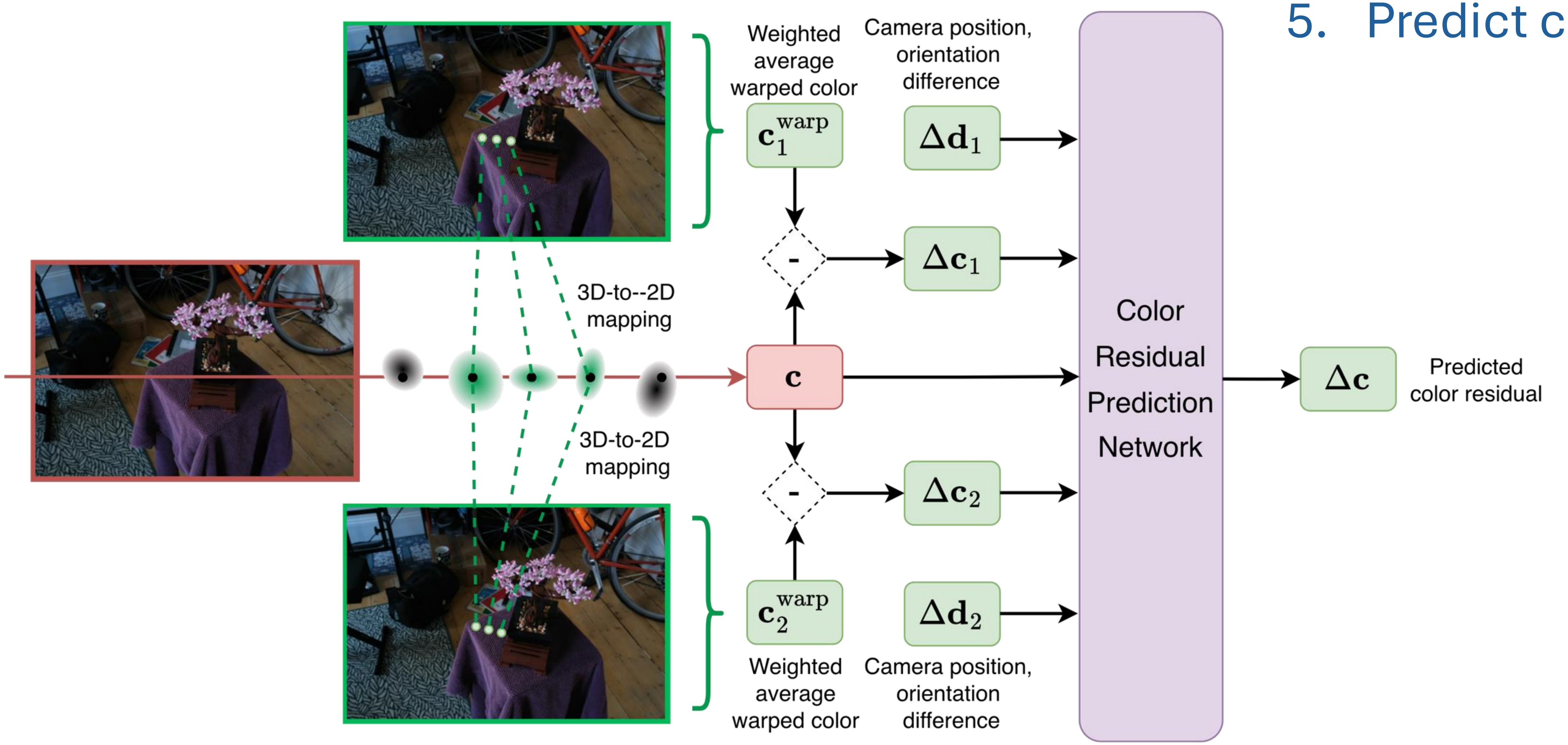
$$\Delta\mathbf{c}_m = \mathbf{c}_m^{\text{warp}} - \mathbf{c}$$

Color Residual Prediction Module



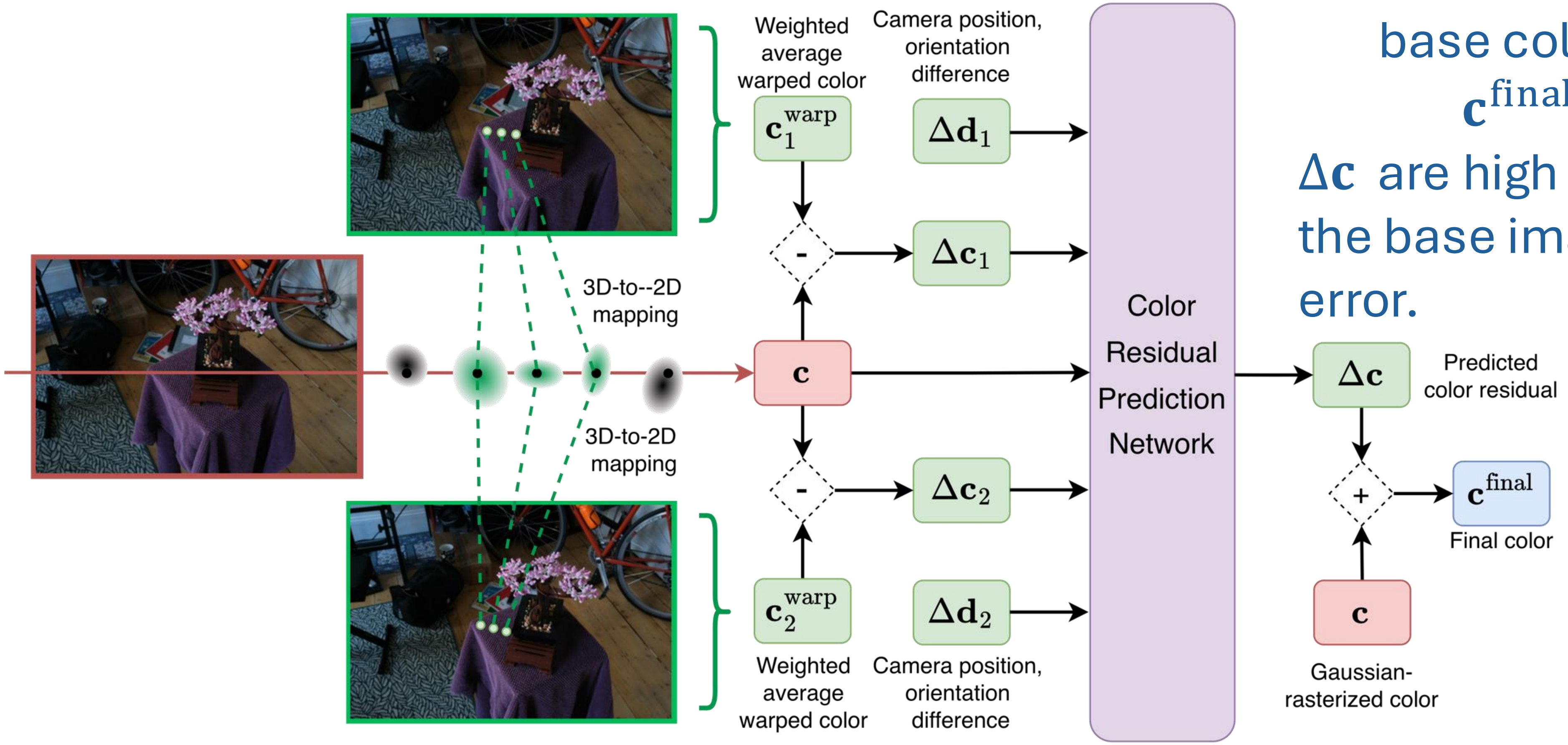
3. Compute the color difference
$$\Delta\mathbf{c}_m = \mathbf{c}_m^{\text{warp}} - \mathbf{c}$$
4. Compute camera pose difference $\Delta\mathbf{d}_m$.

Color Residual Prediction Module



5. Predict color residual $\Delta\mathbf{c}$.

Color Residual Prediction Module



6. Add the residual to the base color.

$$\mathbf{c}^{\text{final}} = \mathbf{c} + \Delta\mathbf{c}$$

$\Delta\mathbf{c}$ are high in regions where the base image $\mathbf{C}$ has large error.

Exposure correction module

- Assume images taken at nearby locations share similar lighting condition.

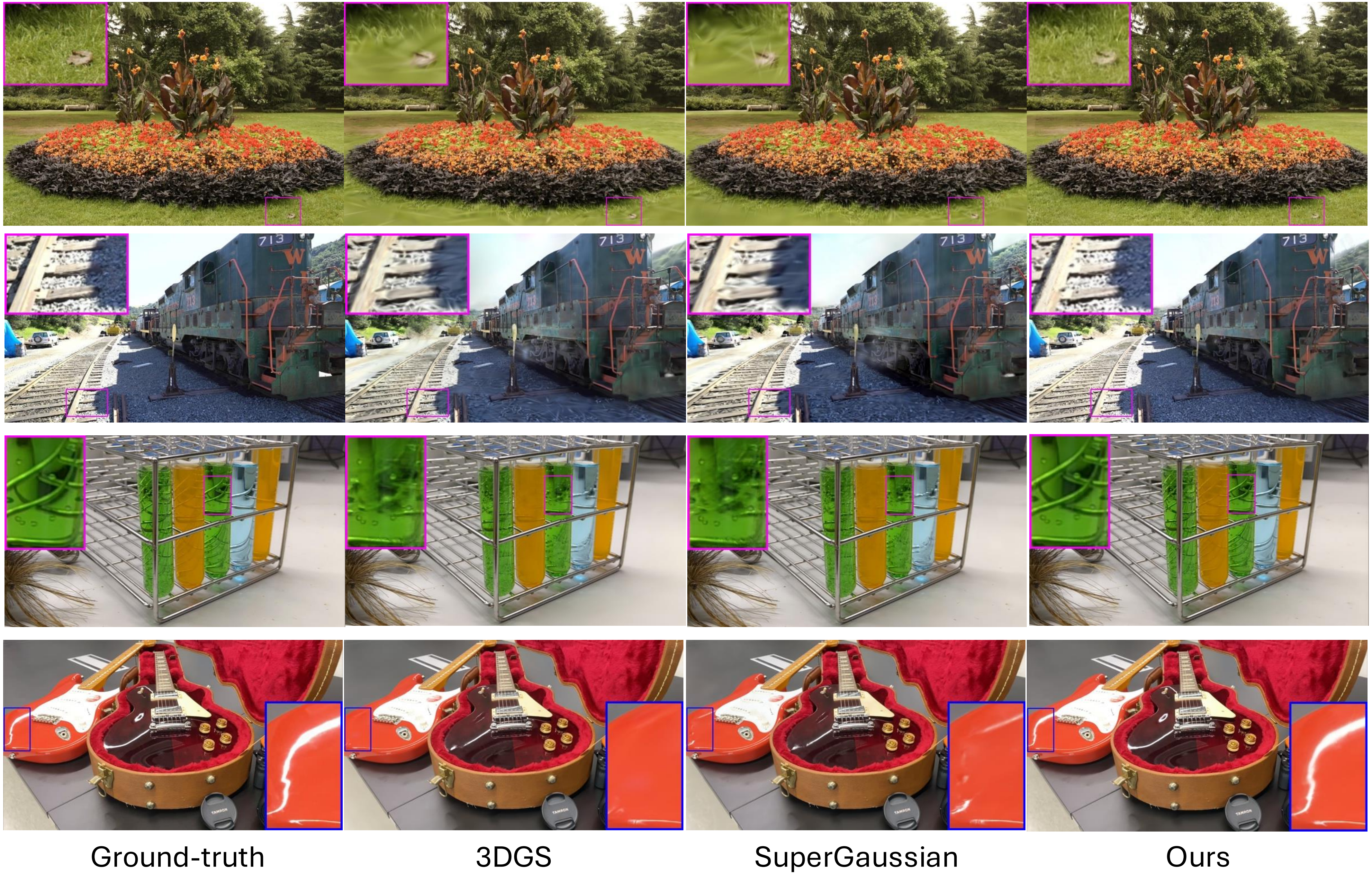
⇒ Correct the exposure of a rendered image by mimicking that of its nearest training view.

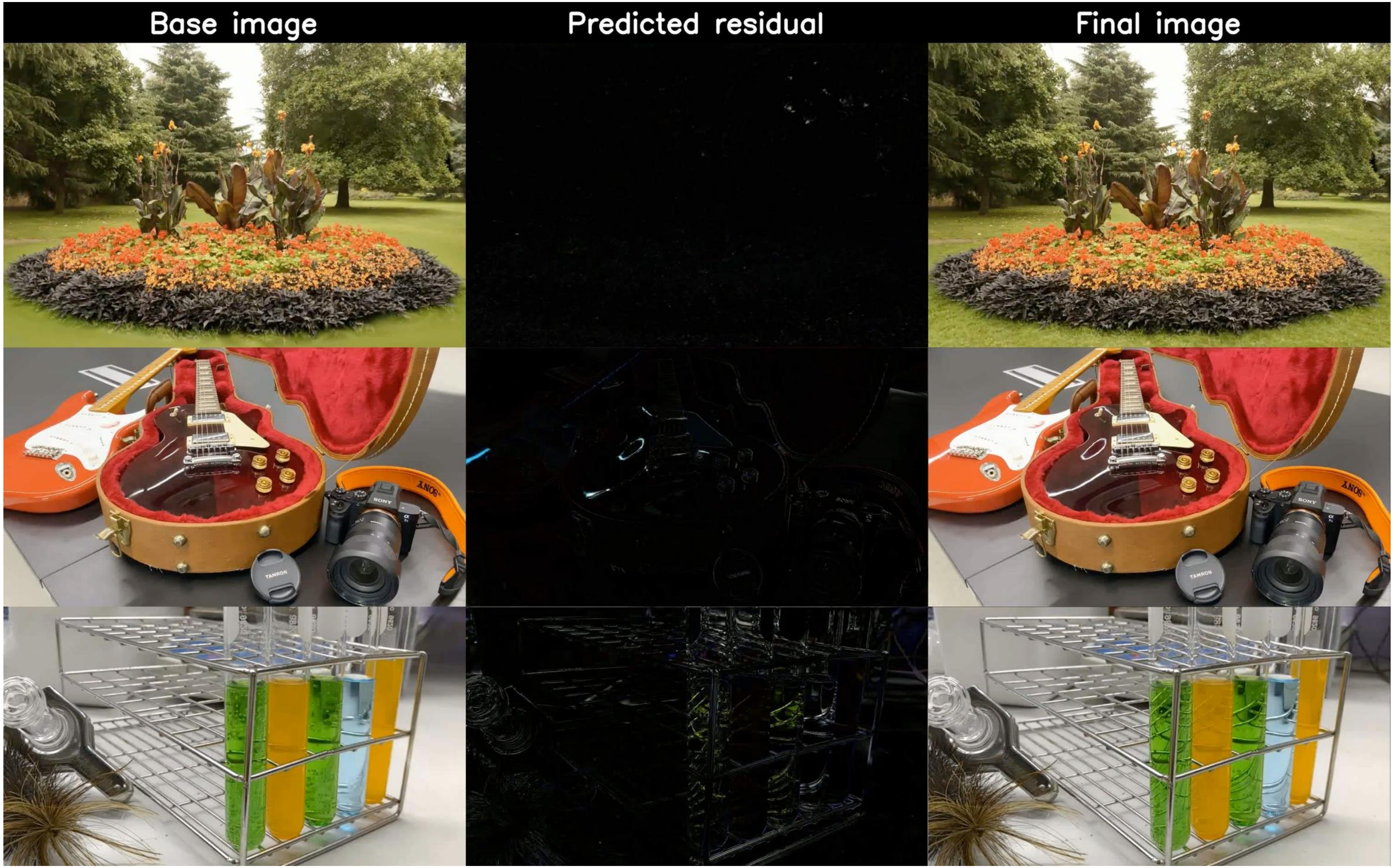
- Find a matrix representing the target view's exposure:

$$\mathbf{A}^* = \arg \min_{\mathbf{A} \in \mathbb{R}^{3\times 4}} \sum_{p \in \chi} \left\| \mathbf{A} \begin{bmatrix} \mathbf{C}(\mathbf{p}) \\ 1 \end{bmatrix} - \mathbf{C}^{\text{near}}(\mathbf{p}) \right\|_2^2$$

- Apply $\mathbf{A}^*$ to correct the exposure of the rendered image:

$$\mathbf{C}^{\text{corr}}(\mathbf{p}) = \mathbf{A}^* \begin{bmatrix} \mathbf{C}(\mathbf{p}) \\ 1 \end{bmatrix}$$


Ground-truth
3DGS
SuperGaussian
Ours


Base image
Predicted residual
Final image

Thanks for watching

Let’s have a chat if you find this work interesting!

Morning session, Thursday Dec 4th

Poster Session 3

Project page