ChromaDistill: Colorizing Monochrome Radiance Fields with Knowledge Distillation
Supplementary Material

Contents

A. Introduction

We present additional results and other details related to our proposed method: ChromaDistill. We present training details in Appendix B.1. We explain the downstream applications in Appendix B.2 and B.3. We present additional experimental results in Appendix C.

B. Implementation Details

B.1. Training Details

We use Plenoxels [11] as neural radiance field representation in our experiments. This representation uses a sparse 3D grid based representation with spherical harmonic (SH) coefficients. For the first stage, luma radiance field, we use the default Plenoxel grid recommended for the type of dataset. We use batch-size of 5000 with RMSProp as optimizer. In the first stage, we use both photometric losses and total-variation (TV) loss proposed in the plenoxels [11]. In the distillation stage, first we get the colorized images from the teacher network. In our experiments, we present result with two image-colorization teachers : 1.) Zhang et al. [56] and 2.) Bigcolor [19]. These colorized images are then used in the distillation stage. When distilling color, we convert the colorized image to "Lab" color space.

Algorithm 2: Color Distillation Algorithm

Input: Trained Nerf Model on Multi-view Grayscale images f_θ, colorization teacher network $\mathcal{T}$

Output: Colorized radiance field network $\mathcal{T}$.

function LOOP(for each image i=1,2.....N do)

$\mathcal{L}_i \leftarrow \phi$

$I_i^C \leftarrow \mathcal{T}(X_i)$.

$I_i^R \leftarrow f_\theta(P_i)$

$\mathcal{L}_i \leftarrow \mathcal{L}_i + \mathcal{L}_{distill}(I_i^C, I_i^R)$

Update f_θ

B.2. Infra-Red Muli-Views

Multi-spectral or Infra-red (IR) sensors are more sensitive to the fine details available in the scene than RGB sensors. Poggi et al. [32] proposed Cross-spectral NeRF (X-NeRF) to model a scene using different spectral sensors. They built a custom rig with a high-resolution RGB camera and two low-resolution IR and MS cameras and captured 16 forward-facing scenes for their experiments. We extracted IR multi-view images and camera poses from the proposed dataset. We naively normalize the IR view between 0 and 1; thus treating it as a grayscale multi-view input sequence. We then apply our method to colorize this view. Our method is effective in colorizing views from different modalities.

B.3. In-the-wild GrayScale Multi-Views

Other than different multi-spectral sensors, there exist lot of in-the-wild grayscale content either in the form of legacy old videos or monochromatic cameras. We extract these multi-view image sequences and then pass these images through COLMAP [35] to extract camera poses. For legacy grayscale image sequences, as there are lot of unnecessary artefacts which affects the perfomrance of COLMAP [35], we pass this sequnce through the video restoration method proposed in [43]. We use the extracted camera-pose and grayscale multi-view image sequence as input for the propsoed method and obtain 3D consistent color-views. This downstream task has a lot of application in Augmented-reality(AR)/Virtual Reality (VR).

B.4. Overview of the Color Distillation Algorithm

Algorithm 2 gives an overview of the color distillation algorithm, For each camera pose, we render a view from the radiance field network trained in grayscale images f_θ. To distill the loss, we colorize the gray-scale teacher using a teacher colorization network

B.5. Notation for "Color Distillation With Multi-Scale Regularization"

- f_θ : NeRF model trained in stage 1 on multi-view grayscale images
- $\mathcal{L}_i$: Loss for i^{th} image in training-set
- $\mathcal{P}_a, \mathcal{P}_a$: Placeholder to save chroma a and b channels from previous scale
- ${}^sI_i^C \leftarrow downsample(I_i^C, 2^s)$: Downsample the image from pre-trained colorization at original resolution by a factor 2^s
- ${}^sI_i^R \leftarrow f_\theta(P_i, s)$: Render an image with the corresponding pose at scale s i.e output width and height be downscaled by a factor 2^s
- $\mathcal{P}_a \leftarrow upsample({}^sa_i^R, 2)$: upsamples the chroma a and b channels for next scale by a factor of 2
- Our method starts from the coarsest scale K i.e image resolution is downscaled by a factor of 2^K

B.6. Colorization Pipeline using Gaussian Splatting [18]

Our proposed knowledge distillation method can be further applied to alternative 3D representations such as Gaussian Splatting [18], which uses rasterization rather than ray-tracing for rendering. We adhere to the default hyperparameters suggested in the original study. Training is conducted only with the luma component up to $15k$ iterations, as Gaussian densification and pruning occur only up to this point. After that, we distill the "a" and "b" channels from the teacher colorization network until $30k$ iterations. We present more qualitative results in Fig. 12. Further, we use a different teacher network for colorization model DdColor [17].

C. Experimental Results

C.1. Grayscale Novel Views

We present quantitative results for generated grayscale novel views from "Luma Radiance Field Stage" (Stage 1) in Table 3. We also compare the generated novel-views with the ground-truth grayscale views in Fig. 13 and 14. We observe that generated novel-views are of good quality. This shows that learning monochromatic signal using a radiance field representation is achievable.

Table 3. Quantitative analysis of GrayScale views

	cake	**pasta**	**buddha**	**leaves**
PSNR	27.772	21.951	23.206	22.146
SSIM	0.855	0.785	0.804	0.784
LPIPS	0.242	0.305	0.347	0.210

Table 4. Characteristic comparison of Our method with Color-NeRF [5]

Method	Extra Parameters	Inference Speed	Supports other 3D representation
Color-Nerf [5]	Yes	High	No
Ours	No	Low	Yes

C.2. Impact of multi-scale regularization

We performed ablation studies on the impact of multi-scale regularization. When distilling color at the original resolution, some areas appeared de-saturated, as seen in the highlighted regions in Fig. 15 (a) & (c). To overcome this issue, we employed multi-scale regularization, which mitigated the color de-saturation during the distillation process. This is evident in the improved color on the grass in playground and on top of the cake, as seen in Fig. 15 (b) & (d). One can observe that a bluish patch is not there with the proposed multi-scale technique. These results demonstrate that our regularization method effectively addresses the color de-saturation problem in the generated views.

C.3. Comparison with Color-NeRF [5]

Color-NeRF [5] is a contemporary work that also solves a similar task. We show additional qualitative results from Color-NeRF in Fig. 16 and 17. We observe that cross-view consistency is not maintained by their method. Further, we compare with the cross-view consistency metrics described in the main paper. Tab. 5 shows that our method performs better short-term and long-term consistency when compared with Color-NeRF. We also draw a comparison of their methodology with ours in Tab. 4. We observe that whereas our method doesnot require any extra parameters to learn color. Color-NeRF requires a separate MLP to learn the color representation. Further, their method is too specific to NeRF architecture. Whereas ours can be easily used with any 3D representation.

C.4. Additional Results

We present additional qualitative results in Fig. 18, Fig. 21 and Fig. 22. We observe that our approach yields 3D consistent color views than the baseline methods. We also present quantitative results in Table 7 and 8. Our method achieves better cross-view consistency compared with the baselines.

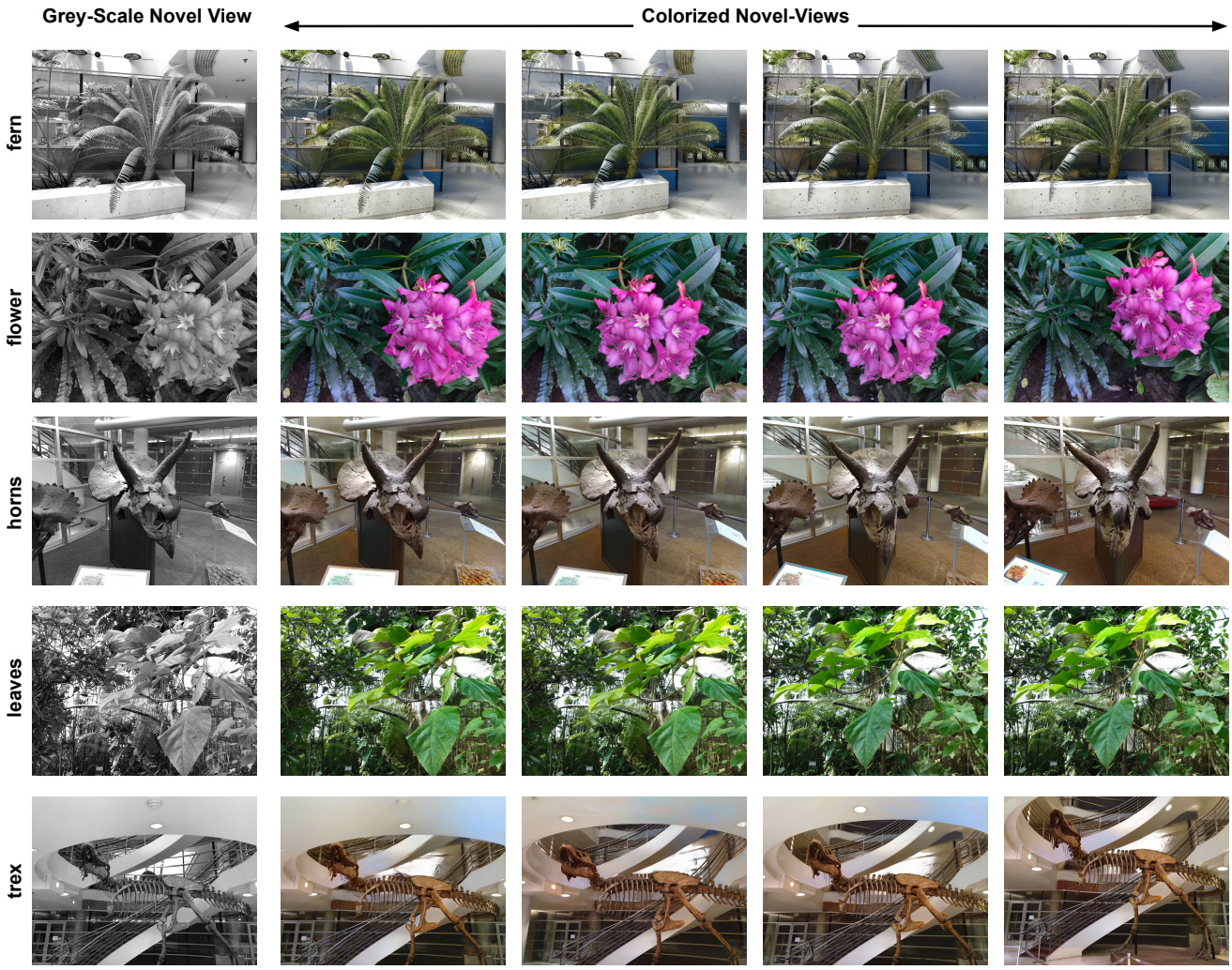

Figure 12. **Colorized Novel-views from Gausssian Splatting 3D representation,** We show results for LLFF scenes : fern, flower, horn, leaves and trex. First column is a grayscale novel-view followed by colorized novel-views using our strategy.

Figure 13. Novel views generated from the input grayscale images for *playground* scene in Tanks & Temples [20] dataset.

C.5. Demonstration on Downstream task.

We show downstream results in Fig. 20. We observe that objects are consistently detected in the colorized novel-views. This downstream task is very useful to enable downstream tasks such as detection for IR sensors.

Table 5. Quantitative comparison of Our method with Color-NeRF [5]. Our method outperforms Color-NeRF for cross-view consistency.

		pasta	**fern**
Short-Term Consistency (↓)	Color-NeRF [5]	0.077	0.021
	Ours	**0.009**	**0.010**
Long-Term Consistency (↓)	Color-NeRF [5]	0.129	0.029
	Ours	**0.017**	**0.011**

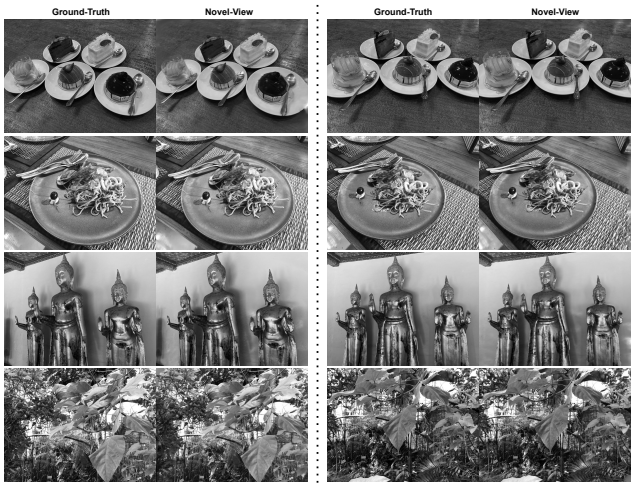

Figure 14. (Top to Bottom) : Comparison of ground-truth and novel-view for grayscale inputs for cake, pasta, buddha and leaves scene.

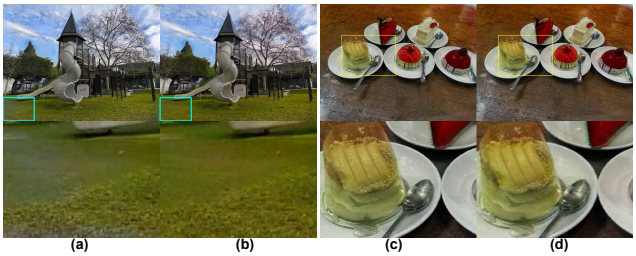

Figure 15. The effect of applying multi-scale regularization on the "playground"((a) and (b)) and "Cake" ((c) and (d)) scene. The highlighted region in the playground (b) and cake (d) had better color in the multi-scale regularization image (than the one w/o multi-scale regularization. Colors in w/o multi-scale regularization are slightly desaturated.

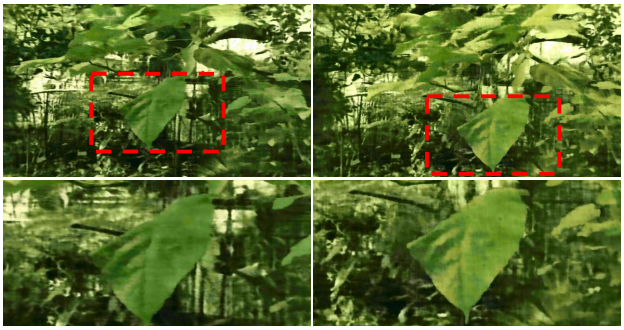

Figure 16. (Top row) Novel-views for "leaves" scene from [5]. (Bottom row) Zoomed in region of the highlighted region. Notice the color change in the leaf.

C.6. Ablation on color-space

We show ablation on color space in Tab. 6, We clearly see better cross-view consistency achieved with "Lab" color space.

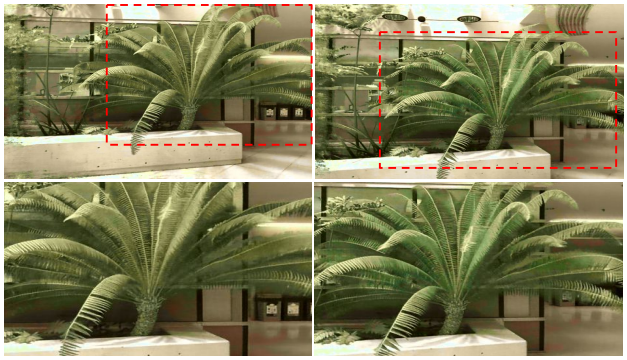

Figure 17. (Top row) Novel-views for "fern" scene from [5]. (Bottom row) Zoomed in region of the highlighted region. Notice that the shade of the fern change from light green to a darker shade of green.

Figure 18. Novel views from the "Different Room", "Fern", and "Ninja bike" scenes are shown in the top, middle, and bottom rows, respectively. Note the consistency across views. To better appreciate these results, please refer to the supplementary video.

Table 6. Ablation results show that using the distillation strategy in the "Lab" color space leads to superior cross-view consistency performance across various scenes.

	Cake	Pasta	Three Buddha	Leaves
Ours(RGB)	0.034	0.027	0.023	0.021
Ours(Lab)	**0.033**	**0.025**	**0.023**	**0.019**

D. Discussion

D.1. Impact of Colorization Teacher Networks.

The proposed method is compatible with any colorization technique. The quality of colorization depends on the selected teacher colorization network. By utilizing BigColor [19] and [56], our approach ensures multi-view consistency irrespective of the chosen teacher network. For instance, in the "cake" scene, [56]. produce dull colors

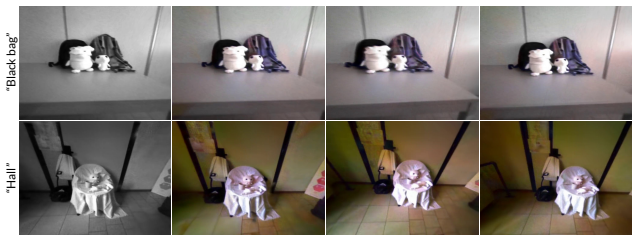

Figure 19. **More IR samples.** (Column 1) Input multi-view IR Sequence. (Columns 2, 3 & 4) Colorized multi-views from Our method. Our approach yields consistent novel-views for a different input modality.

for various objects. In contrast, BigColor generates vivid and sharp colors for different objects. Likewise, we employ DDColor as the teacher colorization network in our infrared experiments.

D.2. Video Colorization Baselines.

Video-colorization methods can generate different colorized outputs for differently rendered trajectories. For example, if we render N videos from N trajectories: $T_1, T_2, ...T_n$ and feed them independently to a video colorization method, this can lead to different outputs even when the same reference images are given. Hence, even though feed-forward video colorization methods can generate temporally consistent views they do not guarantee 3D consistency. Compared to these baselines, our method ensures 3D consistency.

Table 7. Quantitative results for short-term consistency

Scene	BigColor [19] → NeRF	NeRF → DeepRemaster [15]	NeRF → DeOldify [34]	Ours(BigColor [19])
pond	0.022	0.013	0.025	**0.010**
benchflower	0.025	0.013	0.022	**0.010**
chesstable	0.021	0.015	0.022	**0.012**
colorspout	0.025	0.013	0.031	**0.011**
lemontree	0.026	0.015	0.022	**0.014**
stove	0.014	0.010	0.019	**0.008**
piano	0.016	0.010	0.015	**0.009**
redplant	0.029	0.015	0.033	**0.014**
succulents	0.025	0.016	0.027	**0.015**
ninja	0.015	0.011	0.021	**0.007**

Table 8. Quantitative results for long-term consistency

Scene	BigColor [19] → NeRF	NeRF → DeepRemaster [15]	NeRF → DeOldify [34]	Ours(BigColor [19])
pond	0.035	0.017	0.028	**0.015**
benchflower	0.043	0.019	0.030	**0.016**
chesstable	0.033	0.023	0.028	**0.021**
colorspout	0.040	**0.020**	0.051	**0.020**
lemontree	0.041	**0.020**	0.027	0.021
stove	0.018	0.015	0.024	**0.012**
piano	0.026	0.014	0.019	**0.013**
redplant	0.041	0.021	0.041	**0.020**
succulents	0.040	**0.024**	0.032	0.026
ninja	0.021	0.015	0.027	**0.012**

Figure 20. To demonstrate the effectiveness of colorization, we conducted an object detection task on both original infrared (IR) views and their corresponding colorized counterparts. Notably, in (a), no objects are detected in the IR view, and only one out of two objects is detected in (c). However, objects are consistently detected in the colorized views, showcasing the enhanced performance achieved through colorization.

Figure 21. **Qualitative results of our method with baselines.** We display two rows of each scene, each rendered from a different viewpoint. The first four columns depict the original resolution results, while the last four columns show zoomed-in regions of the highlighted areas in the first four columns. The baselines have color inconsistencies in their results, whereas our distillation strategy (columns 4 & 8) maintains color consistency across different views. (Top to bottom) Order of scenes : pond, benchflower, chesstable, colorspout, lemontree

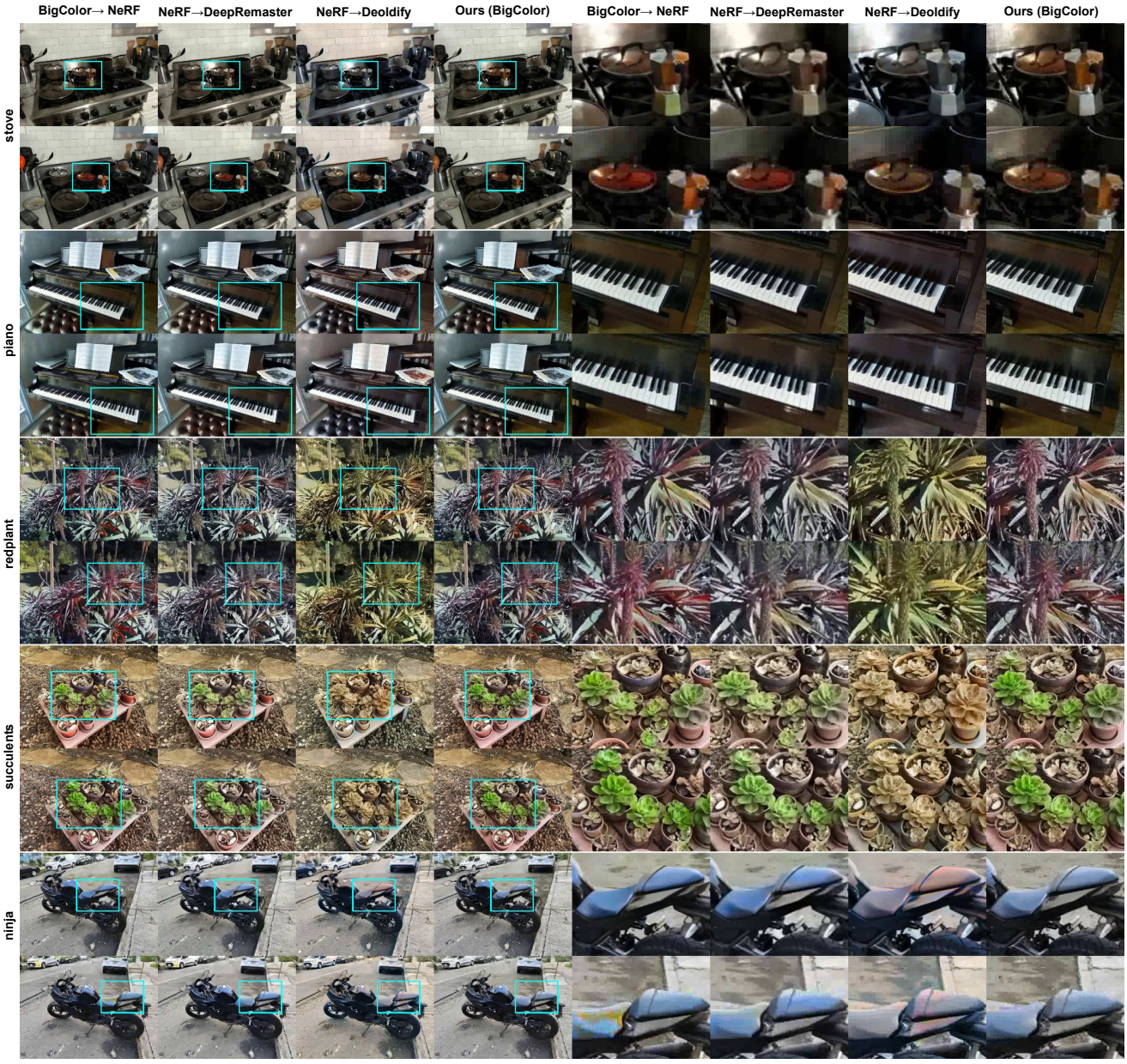

Figure 22. **Qualitative results of our method with baselines.** We display two rows of each scene, each rendered from a different viewpoint. The first four columns depict the original resolution results, while the last four columns show zoomed-in regions of the highlighted areas in the first four columns. The baselines have color inconsistencies in their results, whereas our distillation strategy (columns 4 & 8) maintains color consistency across different views. (Top to bottom) Order of scenes : stove, piano, redplant, succulents, ninja