

000
001
002
003
004
005
006
007
008
009
010
011
012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107

L^AT_EX Author Guidelines for CVPR Proceedings

Anonymous CVPR submission

Paper ID ****

Abstract

The ABSTRACT is to be in fully-justified italicized text, at the top of the left-hand column, below the author and affiliation information. Use the word “Abstract” as the title, in 12-point Times, boldface type, centered relative to the column, initially capitalized. The abstract is to be in 10-point, single-spaced type. Leave two blank lines after the Abstract, then begin the main text. Look at previous CVPR abstracts to get a feel for style and length.

1. Introduction

1.1. Context

Estimating depth from images is an important imaging problem, as dense depth maps are useful precursors for high-level scene understanding tasks like segmentation and pose estimation (cite cite cite) and mid-level perception (of which depth estimation is an important part) has been shown to be very useful for e.g. training robots to navigate their environments. While traditional approaches to depth estimation use multiple cameras or structure-from-motion, convolutional neural networks have also demonstrated reasonable performance on the so-called monocular depth estimation task, where the network is trained to produce a dense depth map given only a single RGB image of the scene.

1.2. Problem Statement

While deep monocular depth estimators have demonstrated strong performance (cite cite) and even some generalizability across scene types, the task they are solving is fundamentally underconstrained due to *inherent scale ambiguity*, i.e. the unresolvable tradeoff between size and distance in monocular images. This ambiguity could be resolved by adding an additional camera and calculating a disparity map, but this method still fails on textureless regions and areas with lots of occlusions. Other approaches use FMCW or time-of-flight LiDAR technologies, but these approaches are currently expensive and bulky.

1.3. Proposed approach

In this paper, we show that by augmenting the RGB image with a histogram of depth information, we can achieve substantially improved performance (and generalizability) over state-of-the-art monocular depth estimators. By performing an exact, weighted histogram matching on the output depth map of the depth estimator, we can match the depth histogram of the scene to the depth histogram of our estimate. Such a histogram can be captured relatively inexpensively using only a single pixel single-photon avalanche diode (SPAD) and pulsed laser illumination diffused over the field of view.

1.4. Impact

Our method is a lightweight postprocessing step that substantially improves the quality of depth maps produced by monocular depth estimators. It can be applied to any method to improve the accuracy instantly. (Our method also helps neural-network-based methods generalize across scene types easily.)

1.5. Limitations

Our method is not without limitations, however. It still requires a laser and single-pixel detector, and as such, is sensitive to ambient photons. Our method, which is a fundamentally a variant of histogram matching, is, like histogram matching, unable to transpose the values of pixels (i.e. if pixel a is farther than pixel b in the input, it will be farther than pixel b in the output). In other words, our method is not able to resolve ordinal depth errors (errors where an object is wrongly placed closer or farther relative to another object). Finally, our method is non-differentiable, and is therefore unsuitable for end-to-end optimization of multi-part networks.

- We introduce the idea of augmenting an RGB camera with a single-pixel SPAD to address scale ambiguity error in monocular depth estimators.
- We analyze our approach on indoor scenes using the NYU Depth v2 dataset. We demonstrate that our ap-

108 approach is able to resolve scale ambiguity while being
 109 fast and easy to implement.
 110

- 111 • (Potentially) We investigate the ability of our method
 112 to help generalization of monocular depth estimators
 113 across scene types. We (hopefully) demonstrate that
 114 our method allows monocular depth estimators to per-
 115 form well even on completely different scene types.
 116
- 117 • We build a hardware prototype and evaluate the effi-
 118 cacy of our approach on real-world data.
 119

2. Related Work

Depth Imaging

- 120 • stereo and multiview
 121
- 122 • structured illumination and random patterns (kinect,
 123 etc.), active stereo
 124
- 125 • time of flight (continuous wave and pulsed)
 126
- 127 • what we do: like pulsed but much simpler setup; no
 128 scanning, no spad array, ...
 129

Monocular Depth Estimation

- 130 • summary of architectures and cost functions: u-net
 131 type architecture with reverse huber loss
 132
- 133 • what we do: same thing, but augment with global hints
 134 (inspired by these approaches, we do ...)
 135

136 **Deep Sensor Fusion** global hints for super-resolution,
 137 colorization, depth estimation
 138

- 139 • colorization
 140
- 141 • david's 2018 paper for depth estimation and denoising
 142 (see david's 2019 sig paper for related work)
 143
- 144 • what we do: slightly different application
 145

146 **Histogram Matching** Histogram matching as an image
 147 processing technique
 148

- 149 • Exact histogram matching paper used in this work
 150
- 151 • Wasserstein-based optimization techniques for
 152 histogram-based regularization
 153

3. Method

154 In this section, we describe the measurement model for a
 155 single-pixel time-of-flight lidar sensor under diffuse, pulsed
 156 laser illumination.
 157

3.1. Measurement Model

158 Consider a laser which emits a pulse at time $t = 0$ with
 159 time-varying intensity $g(t)$ uniformly illuminating some 3D
 160 scene. We parameterize the geometry of the scene as a
 161 height map $z(x, y)$. Neglecting albedo and falloff effects,
 162 an ideal detector counting photon events from a location
 163 (x, y) in the time interval $(n\Delta t, (n + 1)\Delta t)$ would record
 164

$$\lambda_{x,y}[n] = \int_{n\Delta t}^{(n+1)\Delta t} (f * g)(t - 2z(x, y)/c) dt \quad (1)$$

165 where c is the speed of light, and f is a function that
 166 models the temporal uncertainty in the detector. Single-
 167 photon avalanche diodes (SPADs) are highly sensitive photodetectors
 168 which are able to record single photon events
 169 with high temporal precision [?]. Since the detection of
 170 each photon can be described with a Bernoulli random variable,
 171 the total number of accumulated photons in this time
 172 interval follows a Poisson distribution according to
 173

$$h[n] \sim \mathcal{P}\left(\sum_{x,y} \alpha_{x,y} \eta \lambda_{x,y}[n] + b\right) \quad (2)$$

174 where $\alpha_{x,y} = r_{x,y}/z(x, y)^2$ captures the attenuation of
 175 the photon counts due to the reflectance $r(x, y)$ of the scene
 176 and due to the inverse square falloff $1/z(x, y)^2$. In addition,
 177 η is the detection probability of a photon triggering a SPAD
 178 event, and $b = \eta a + d$ is the average number of background
 179 detections resulting from ambient photons a and erroneous
 180 “dark count” events d resulting from noise within the SPAD.
 181

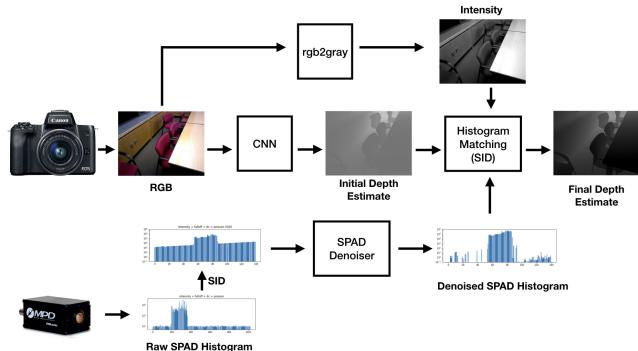


Figure 1: **Overview of the full pipeline** We use a CNN to get an initial per-pixel depth estimate. We then perform gradient descent to optimize that estimate using the SPAD forward model and the dual-Sinkhorn distance .

3.2. Monocular depth estimation with global depth hints

Given a single RGB image $I(x, y)$ and a vector of photon arrivals $h[n]$ described by equation 2, we seek to reconstruct the ground truth depth map $z(x, y)$. Our method has two parts. First, we initialize our estimate of the depth map from the single RGB image via a monocular depth estimator described below. Second, we refine this depth map using the captured measurements $h[n]$ via a process we call Differentiable Histogram Matching (DHM). Differentiable histogram matching is a tool for post-processing the image to match the depth map to the statistics we capture from the SPAD.

Initialization via CNN Convolutional Neural Networks have become increasingly capable of leveraging monocular depth cues to produce accurate estimates of depth from only a single image. We therefore choose to initialize our depth map estimate $\hat{z}^{(0)}(x, y)$ using a CNN. However, any depth estimator reliant on only a single view will be unable to resolve the inherent scale ambiguity in the scene resulting from the tradeoff between size of and distance to an object. The next step, differentiable histogram matching, will resolve this ambiguity using the depth information present in the SPAD histogram.

SPAD Denoising

- Discuss MLE for SPAD denoising
- Write optimization problem for SPAD denoising
- Show performance on a few examples

Exact Histogram Matching An image’s *histogram* is a pair of vectors (h, b) where h_i is the number of pixels of the image whose value lies in the range $[b_i, b_i + 1)$. Then, given a source image S with histogram (h_s, b) and a target histogram (h_t, b) , histogram matching generates a new image M such that $h_m \approx h_t$ and the pixel values in M are in the same relative order as in S .

3.3. Implementation Details

For the Monocular Depth Estimator, we use pretrained versions of the the Deep Ordinal Regression Network (DORN) [] and the DenseDepth Network. The exact histogram matching method is as described in [].

4. Simulation

4.1. Implementation Details

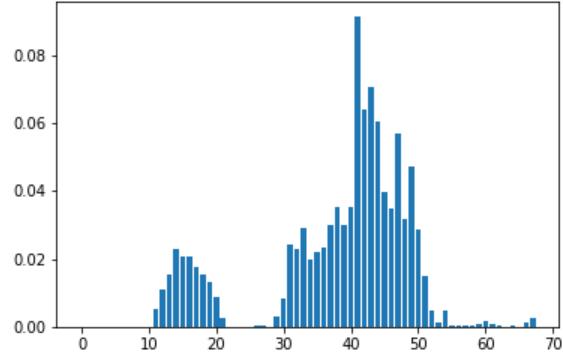
- Number of bins used, depth range, laser parameters, use of intensity image.
- Using

NYU Depth v2 The NYU Depth v2 Dataset consists of 249 training and 215 testing scenes of RGB-D data captured using a Microsoft Kinect. We used a version of DORN pre-trained according to [?] as our CNN.

324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394

340

Raw Depth Histogram



SPAD Counts (Normalized)

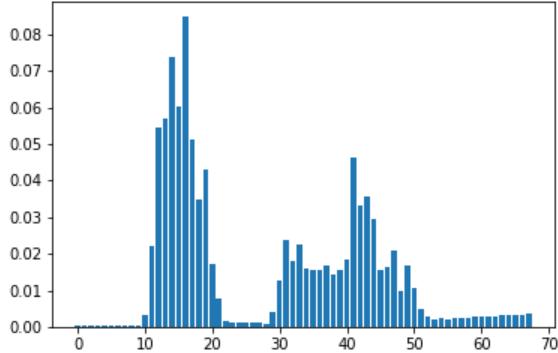
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355

Table 1: Sample Image. Top Left is the RGB image. Top Right is ground truth depth. Bottom Left is Raw ground truth depth histogram. Bottom Right is simulated SPAD measurements. Notice how closer depths are magnified and far depths are attenuated.

358

359
360
361

5. Hardware Prototype

362
363
364
365
366
367

5.1. Setup

- Description of hardware used
- Images of scenes used

368
369
370
371
372
373
374
375
376
377

6. Discussion

422
423
424
425
426
427
428
429
430
431

	$\delta^1 \uparrow$	$\delta^2 \uparrow$	$\delta^3 \uparrow$	$rel \downarrow$	$rmse \downarrow$	$log10 \downarrow$	
432							486
433							487
434							488
435							489
436							490
437							491
438							492
439							493
440							494
441							495
442							496
443							497
444							498
445							499
446							500
447							501
448							502
449							503
450							504
451							505
452	DORN	0.846	0.954	0.983	0.120	0.501	0.053
453	DORN + median rescaling	0.871	0.964	0.988	0.111	0.473	0.048
454	DORN + GT histogram matching	0.906	0.972	0.990	0.095	0.419	0.040
455	Proposed (SBR=10)	<u>0.903</u>	0.970	<u>0.989</u>	0.091	0.422	<u>0.040</u>
456	Proposed (SBR=50)	0.906	<u>0.971</u>	0.990	0.089	<u>0.410</u>	0.039
457	Proposed (SBR=100)	0.906	<u>0.971</u>	0.990	<u>0.090</u>	0.408	0.039
458	DenseDepth	0.847	0.973	0.994	0.123	0.461	0.053
459	DenseDepth + median rescaling	0.888	0.978	0.995	0.106	0.409	0.045
460	DenseDepth + GT histogram matching	0.930	0.984	0.995	0.079	0.338	0.034
461	Proposed (SBR=10)	0.922	0.982	<u>0.994</u>	0.082	0.361	0.036
462	Proposed (SBR=50)	0.925	<u>0.983</u>	0.995	<u>0.081</u>	0.348	<u>0.035</u>
463	Proposed (SBR=100)	0.926	<u>0.983</u>	0.995	<u>0.081</u>	0.346	<u>0.035</u>

Table 2: Simulated Results on NYU Depth v2. Bold indicates best performance for that metric, while underline indicates second best.

464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485

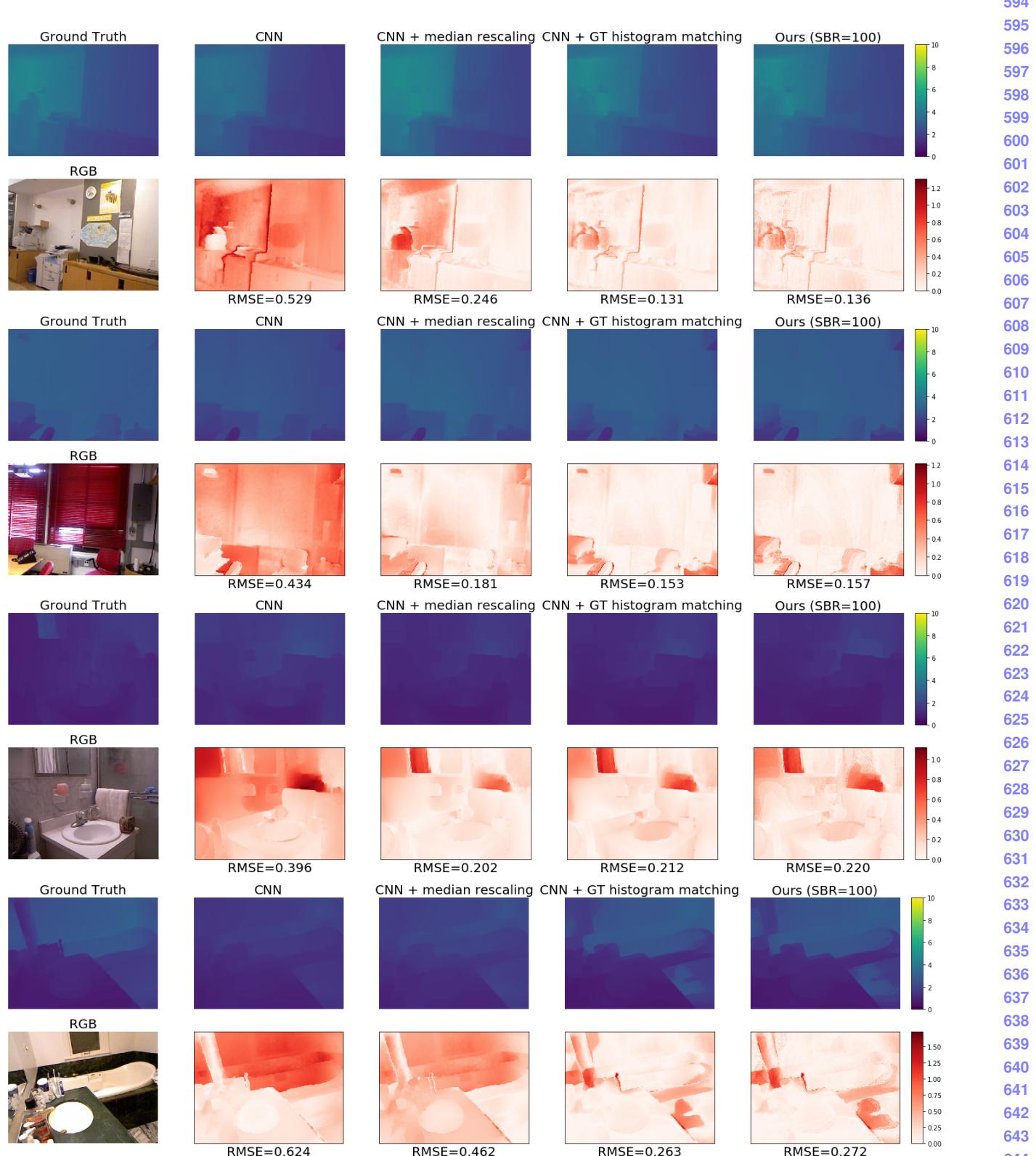


Figure 2: Results on DenseDepth

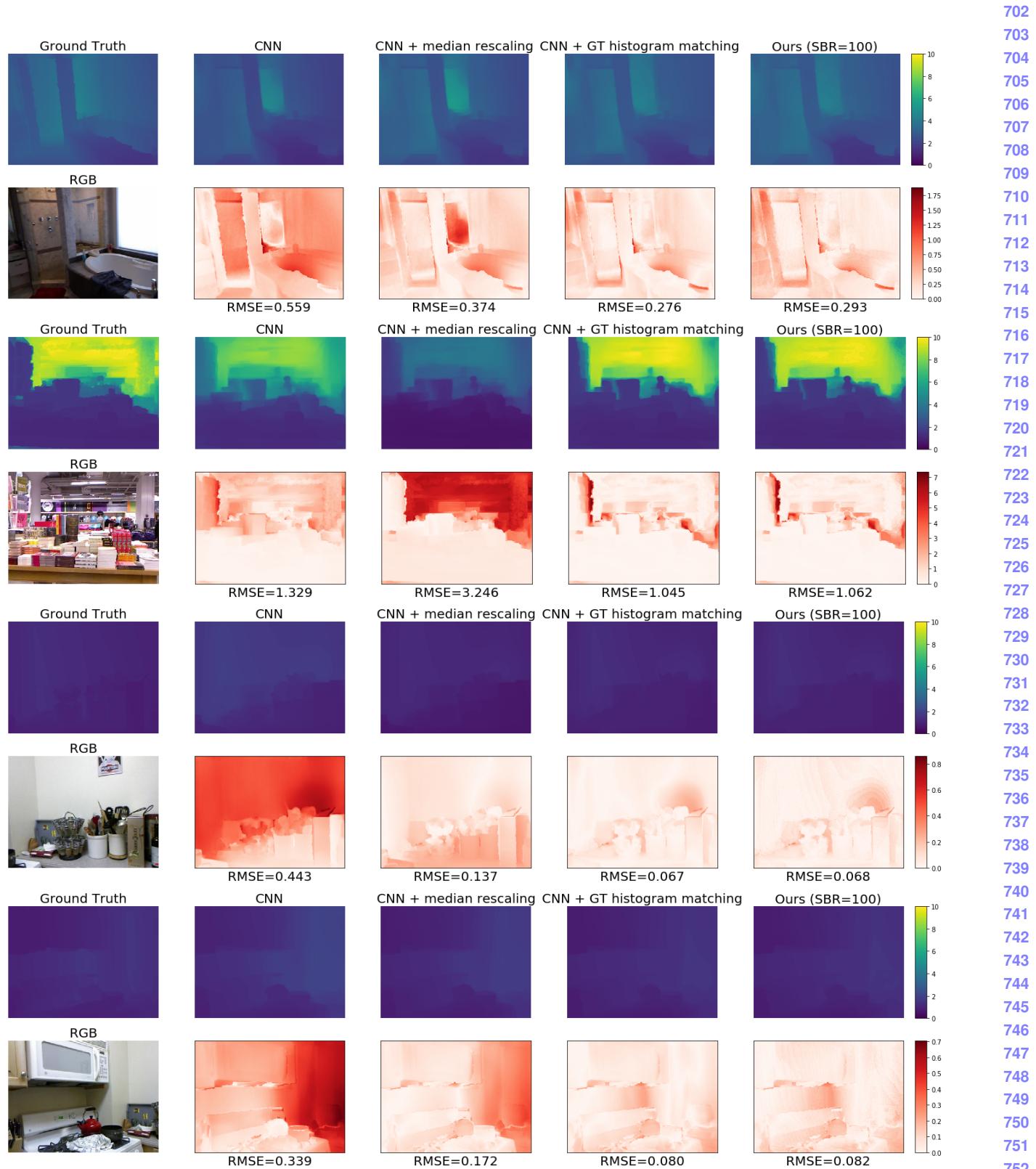


Figure 3: Results on DenseDepth

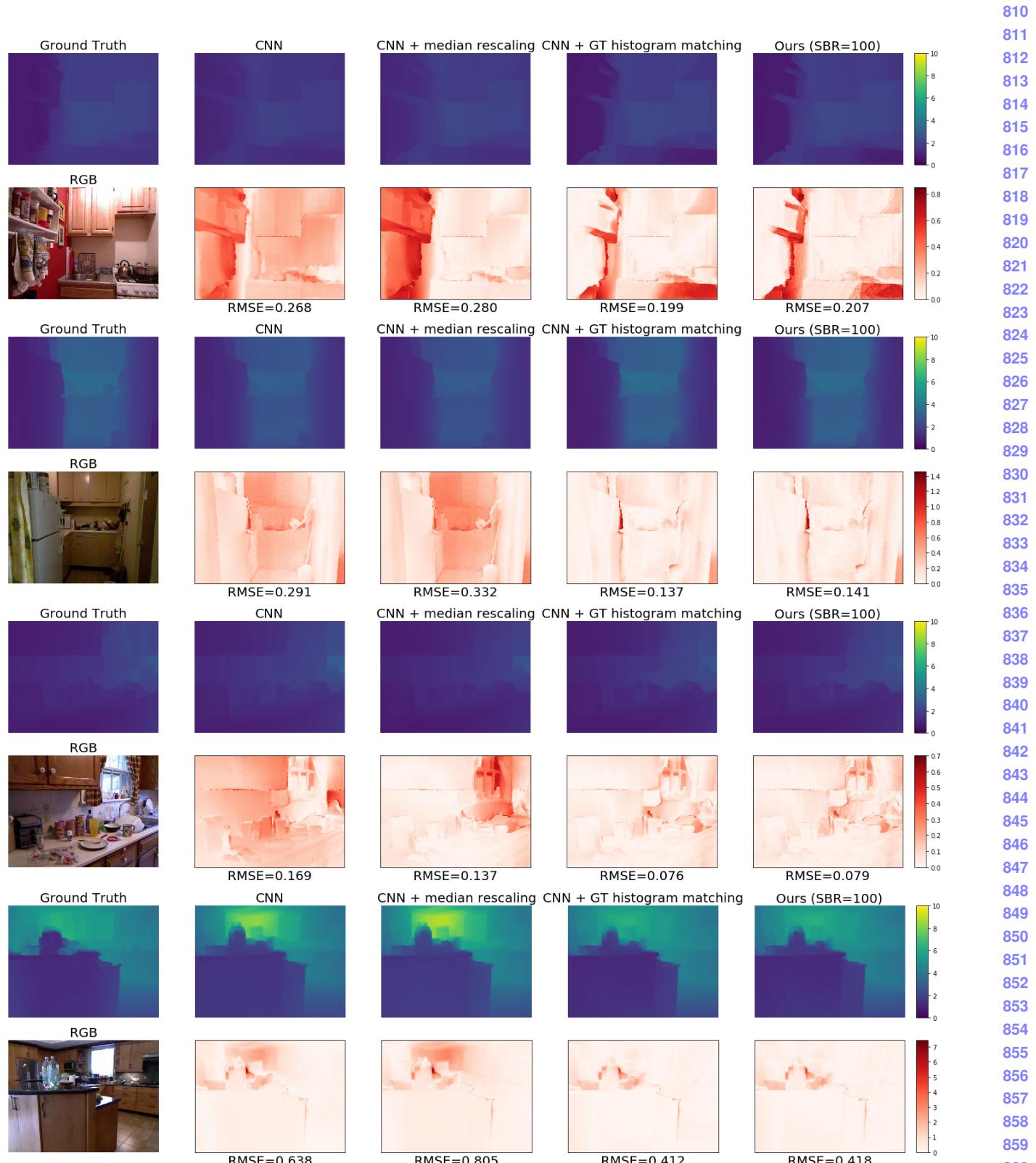


Figure 4: Results on DenseDepth

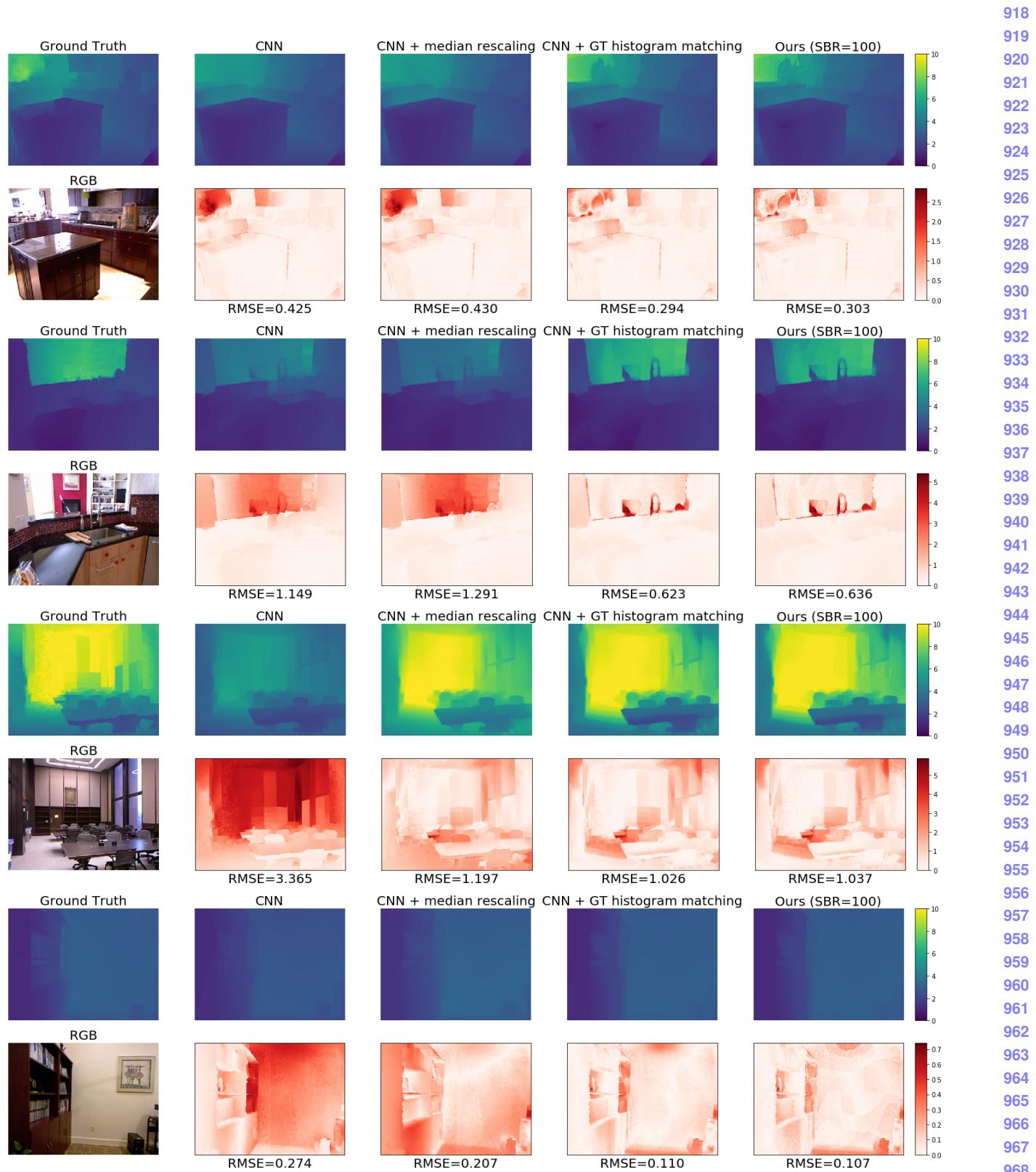


Figure 5: Results on DenseDepth

CVPR
#****

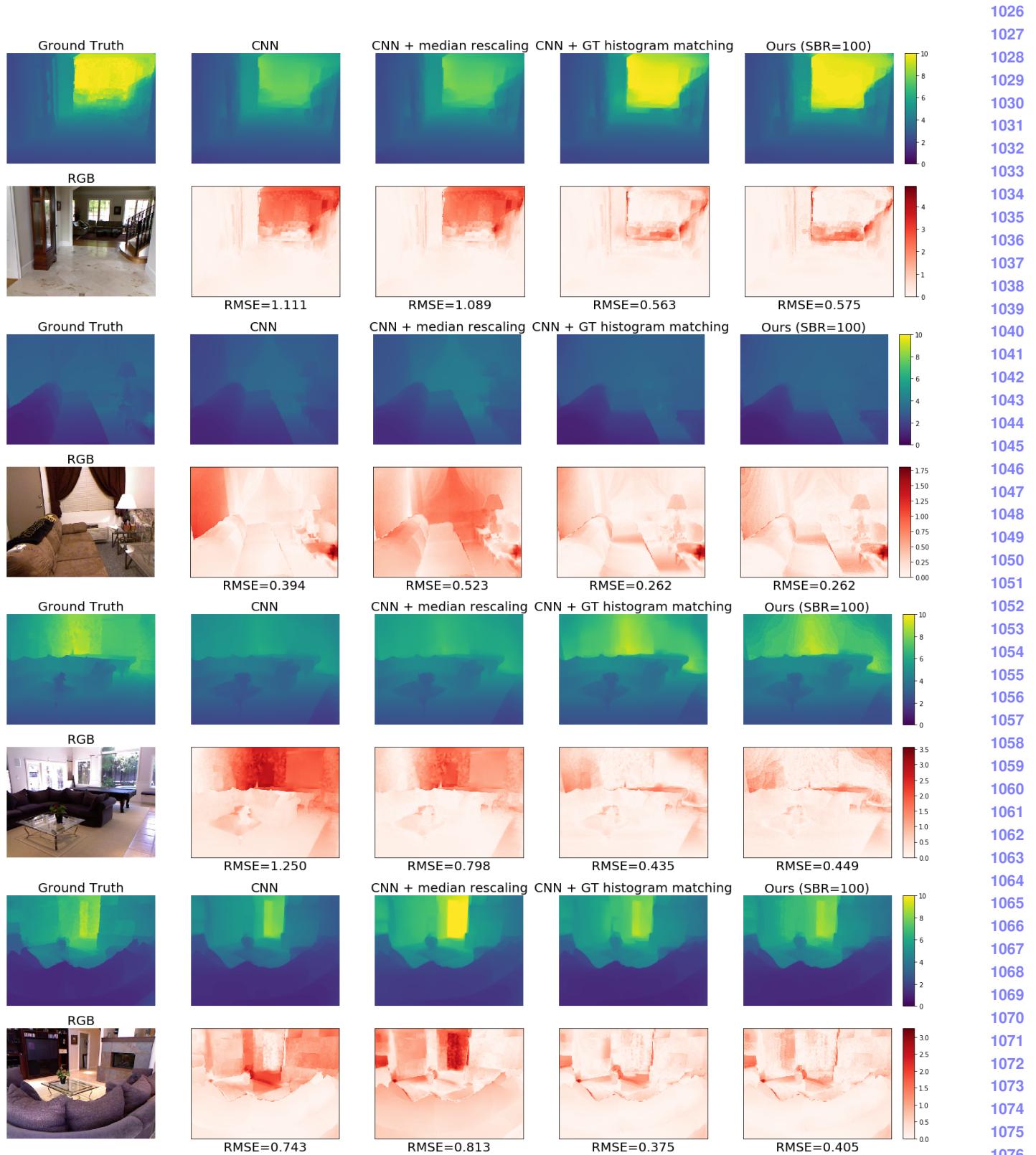


Figure 6: Results on DenseDepth

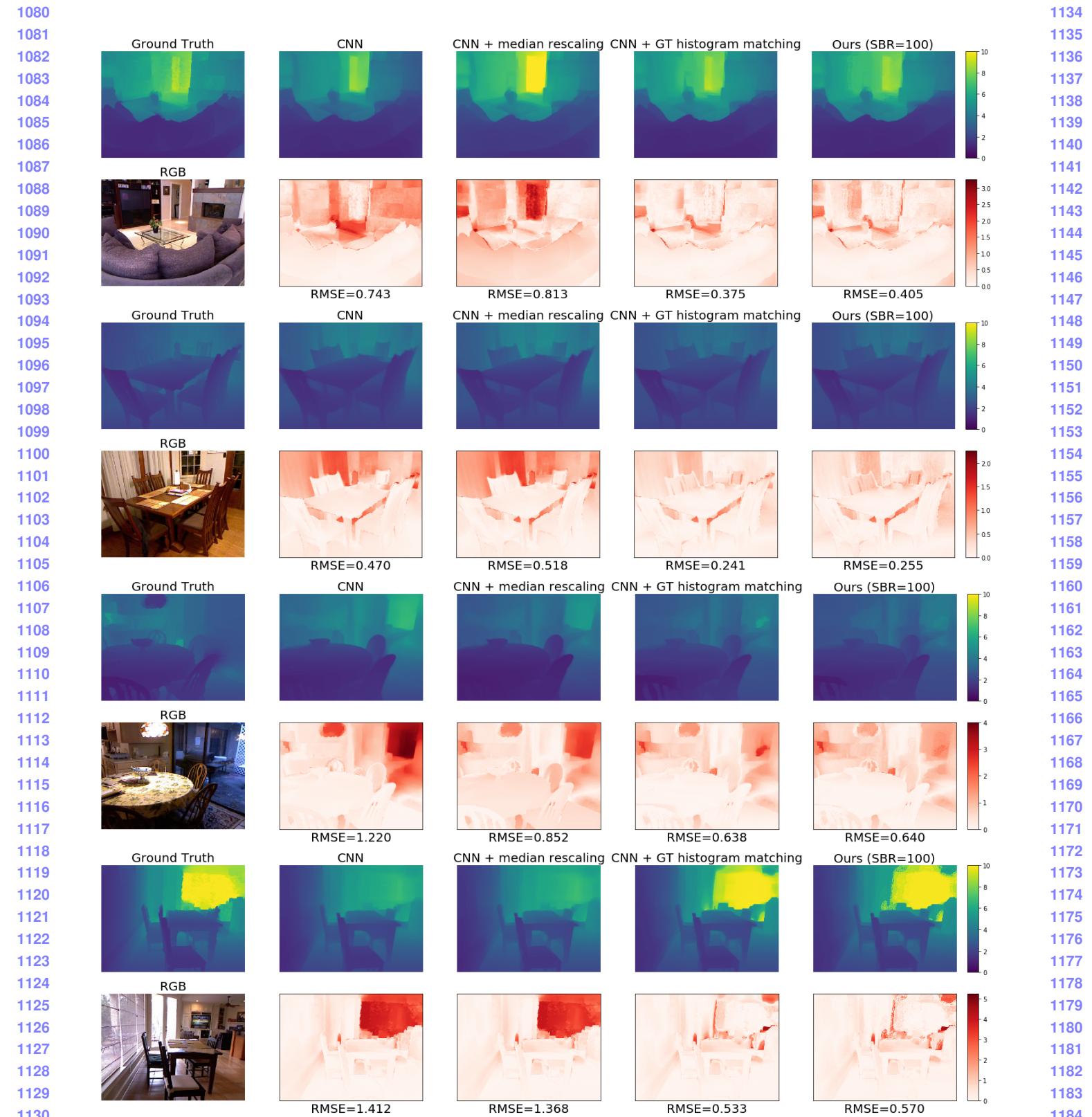


Figure 7: Results on DenseDepth

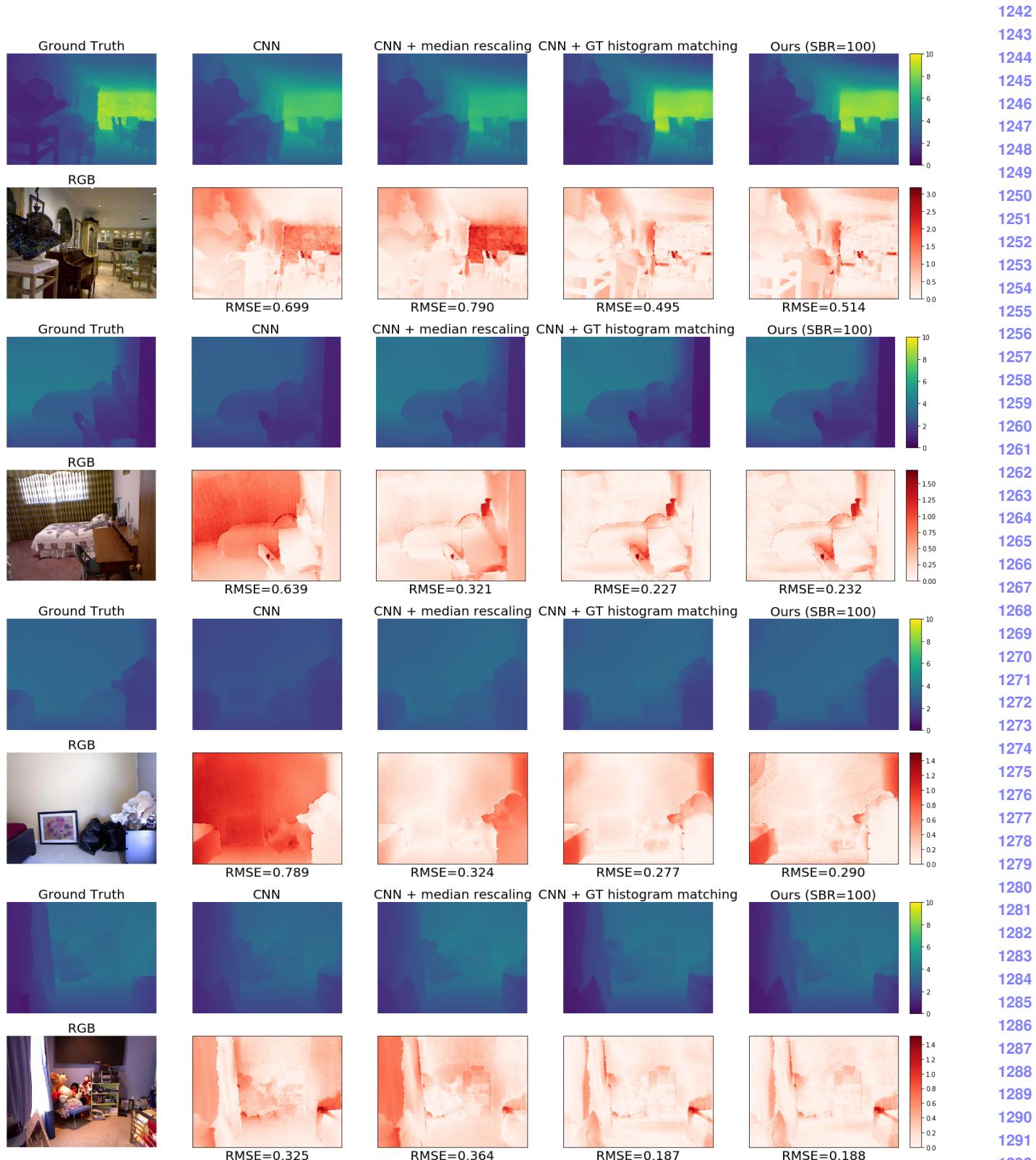


Figure 8: Results on DenseDepth

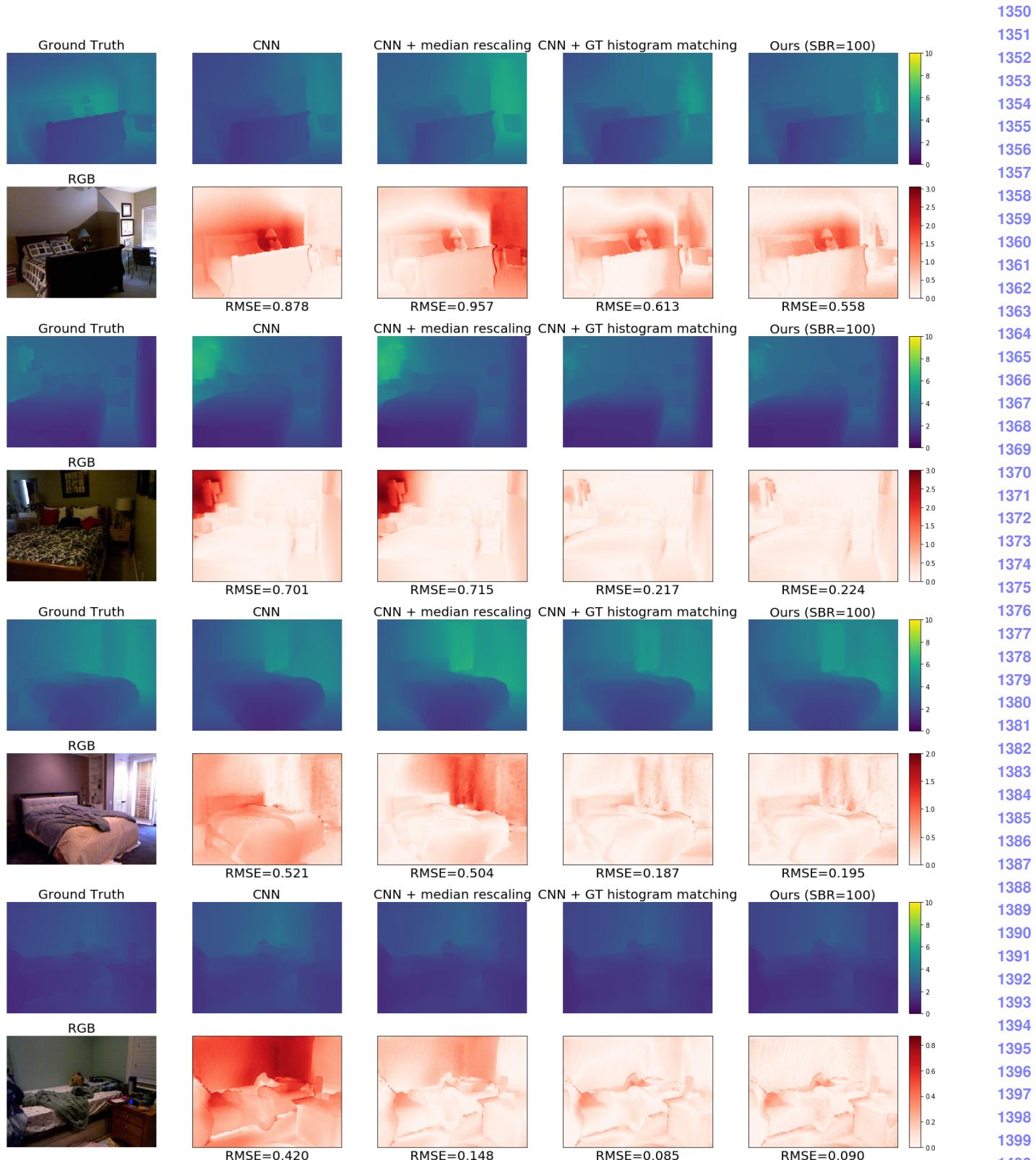


Figure 9: Results on DenseDepth

1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433

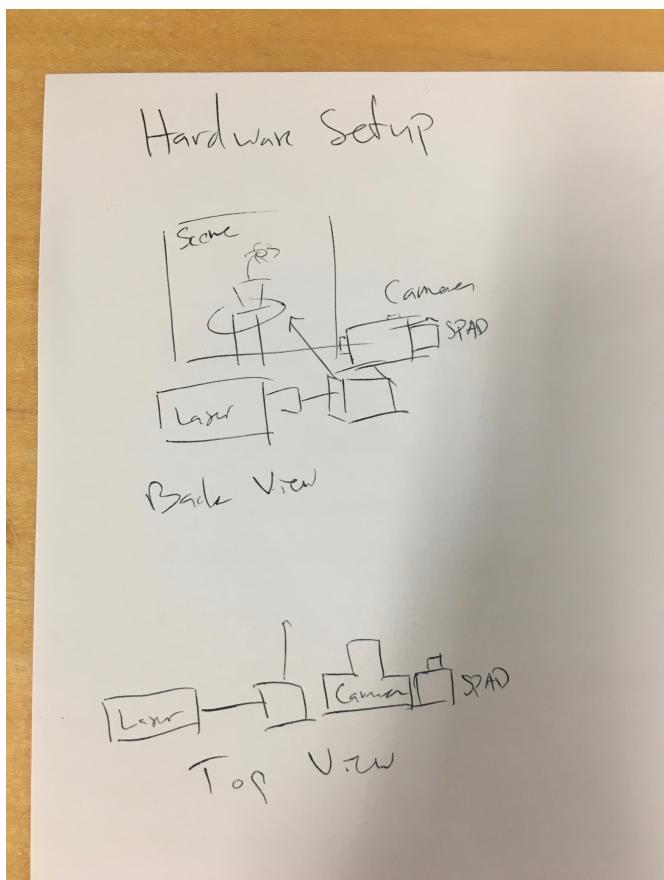


Figure 10: **Hardware setup**

1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457

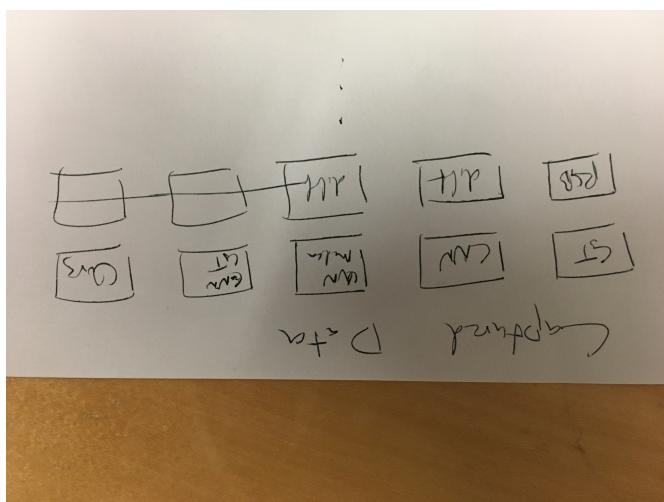


Figure 11: **Hardware results**

1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511