Single View Metrology in the Wild: Supplementary Material

Rui Zhu$^{1[0000-0002-3266-2514]}$, Xingyi Yang$^{1[0000-0002-1603-9829]}$, Yannick Hold-Geoffroy$^{2[0000-0002-1060-6941]}$, Federico Perazzi$^{2[0000-0002-3636-8267]}$, Jonathan Eisenmann$^{2[0000-0003-2018-0793]}$, Kalyan Sunkavalli$^{2[0000-0002-6030-2348]}$, and Manmohan Chandraker$^{1[0000-0003-4683-2454]}$

[1] University of California San Diego, La Jolla CA 92093, USA
{rzhu,mkchandraker}@eng.ucsd.edu
[2] Adobe Research, San Jose CA 95110, USA
{holdgeof,perazzi,eisenman,sunkaval}@adobe.com

1 Table of Contents

- In Section 2 we list details of the implementation and model, as well as some parameters in training;
- In Section 3 we give the formulation of the perspective camera model;
- In Section 4 we evaluate the camera calibration performance of our model against [2];
- In Section 5 we introduce the data preparation details, examples and statistics of all datasets;
- In Section 6 we list extra results for all datasets, as well as corner cases and failure cases on COCO-Scale. Additionally, we include samples to support our conclusion in the main paper, that (1) by introducing more objects from multiple categories strengthens the results (Table 4 of the main paper), and (2) our model which explicitly accounts for keypoints and upright ratio better deals with person in sitting pose (Table 6 of the main paper).

2 Network Architecture and Training Details

In this section, we provide additional details on the neural network and its training process.

Our network is adapted from the `e2e_keypoint_rcnn_R_50_FPN_1x_caffe2` meta-architecture of `maskrcnn-benchmark` [5]. We use their proposed architecture for our backbone, FPN, ROI bounding box head, and keypoint head. In addition to those network heads, we add an object height head, and a camera calibration head (including a separate camera height head). Throughout this section, we denote a convolution layer as *Conv(C, K, S)* and fully-connected layer as *FC(C)* where C is the number of output channels, K is the kernel size, and S is the stride.

The camera calibration head is adapted from the feature extractor of FPN2MLP and FCPredictor. The output of this head consists of 256 bins, on which a Softmax operation is applied to obtain the estimation.

The object height estimation head shares the same feature extractor with the keypoint head. This object height estimation head consists of three layers: *Conv(3, 256, 2)*, *FC(1024)* and *FC(256)*. We use the ReLU activation function for all layers except the output layer, which produces logits.

The network is trained in two stages. In the first stage, the backbone and camera calibration head, keypoint head and bounding box head are trained jointly on the Calib dataset (for camera parameters) as well as COCO-Scale (for keypoints and bounding boxes). To train this stage, we employ the following combination of losses:

$$\mathcal{L}_{\text{first}} = +\alpha_3 \mathcal{L}_{\text{calib}} + \alpha_4 \mathcal{L}_{\text{det}} + \alpha_5 \mathcal{L}_{\text{kps}} \ , \tag{1}$$

where $\alpha_3 = 1$ and $\alpha_4 = \alpha_5 = 10$. We use SGD as optimizer with a learning rate of 1e-5, and a batch size of 16. The training takes 40k iterations to converge.

In the second stage, the previous modules are further trained solely on COCO-Scale with the bounding box fitting v_t and the object prior y_{h_j}. To train the second stage, we add this bounding box fitting loss to the first stage loss, resulting in the following training loss:

$$\mathcal{L}_{\text{second}} = \alpha_1 \sum_{j=1}^{M} \mathcal{L}_{v_{t_j}} + \alpha_2 \sum_{j=0}^{M} \mathcal{L}_{h_{\text{obj}_j}} + \alpha_3 \mathcal{L}_{\text{calib}} + \alpha_4 \mathcal{L}_{\text{det}} + \alpha_5 \mathcal{L}_{\text{kps}}, \tag{2}$$

where $\alpha_1 = 1$, $\alpha_2 = 0.05$, $\alpha_3 = 1$ and $\alpha_4 = \alpha_5 = 10$. The training is performed with the Adam optimizer using an initial learning rate of 1e-5. The batch size is 16 with 4 examples from Calib and 12 from COCO-Scale. The training takes 120k iterations to converge.

2.1 Other Parameters

All classification outputs of the network uses the Softmax operation on 256 output bins following [2]. Those bins represent different ranges of values according to their task. The range for camera height h_{cam} is [0.5, 5] for the initial prediction, and [-0.3, 0.3] for refinement layers. For the camera parameters, the range is [-0.6, 0.6] for pitch θ, and [0.2389, 1.6] rad for field of view h_θ. For people height h_{obj}, the bins are [1., 1.9], [-0.3, 0.15], [-0.10, 0.10], and for cars are [1.40, 1.70], [-0.10, 0.10], [-0.05, 0.05], for the initial prediction and refinement layers, respectively.

For data pre-processing, we use identical pipeline as in the e2e_keypoint_rcnn_R_50_FPN_1x_caffe2 meta-architecture of maskrcnn-benchmark [5].

3 Camera Model Details

This section describes the image formation model used in our method.

Assuming a pinhole camera model, a point $\mathbf{x} = [x, y, z]^T \in \mathbb{R}^3$ in camera coordinates and its reprojected pixel coordinates in the image frame $\mathbf{p} = [u, v]^T \in \mathbb{R}^2$, the camera projection function can be written as

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & u_c \\ 0 & f & v_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & -h_{\text{cam}} \\ 0 & \sin\theta & \cos\theta & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \tag{3}$$

where $[u_c, v_c] \in \mathbb{R}^2$ is the optical center which we assume are known, and the output $[u, v, w]$ on the left-hand side is in homogeneous coordinates.

Based on this perspective camera projection model, we may derive v_t and v_b according to Eqn. 2 and Eqn. 3 of the main paper.

4 Camera Calibration Evaluation on Calib Dataset

We evaluate camera calibration results of our model against Hold-Geoffroy *et al.* [2] on the validation split of the Calib dataset. While it is not our central goal, we achieve strong camera calibration results compared to [2] under the same assumptions of pre-corrected images or minimal lens distortion, as included in Table 1.

Table 1: Camera calibration error on Calib, shown in mean ± std. Best results in bold

	pitch $\theta(^\circ)$	roll $\psi(^\circ)$	field of view $h_\theta(^\circ)$
Hold-Geoffroy *et al.* [2]	2.11±3.10	1.19±1.89	4.39±3.67
SN-L3	1.83±2.64	**1.02±1.46**	**3.61±3.21**
SN-L3-kps-can	**1.82±2.62**	1.05±1.94	3.63±3.22

5 Dataset Details

5.1 COCO-Scale Dataset

Data Pruning We use some filtering rules to prune COCO [4] into our COCO-Scale. See Fig. 1 for an example of those filtering rules. We use the provided “stuff” annotations as non-object areas. On Fig. 1 (right), we show 14 detected people from Mask R-CNN. For detection, we apply 4 rules:

1. the bounding box width-to-height ratio should be within [2.0, 8.0] for people, and [1/3.5, 1/1.5] for cars;

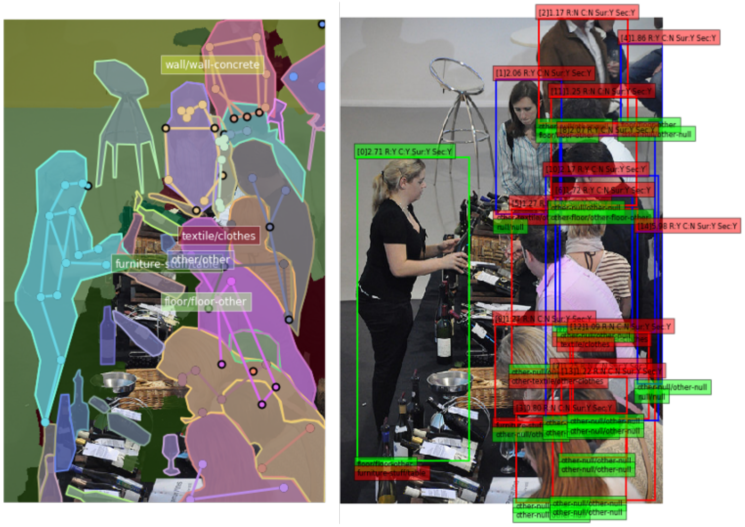
wall/wall-concrete
textile/clothes
other/other
furniture-stuff/table
floor/floor-other
[0]2.71 R:Y C:Y Sur:Y Sec:Y
[2]1.17 R:N C:N Sur:Y Sec:Y
[4]1.86 R:Y C:N Sur:Y Sec:Y
[1]2.06 R:Y C:N Sur:Y Sec:Y

Fig. 1: (Left) Stuff annotation and human keypoints annotation from COCO. (Right) Examples of our filtering rules. People highlighed in red are discarded due to their abnormal width-to-height ratio, usually caused by incomplete observation. People highlighted in blue are flagged by another filtering rule (amodal observation, non-standing pose, supporting surface). The green bounding box means the person passed all our filtering rules. For each bounding box, we display in the upper-left corner the object index, its width-to-height ratio, R:Y/N (for ratio), C:Y/N (for amodal, *i.e.* the head and ankles are both visible), Sur:Y/N (lying on a valid surface), Sec:Y/N (no major occlusion). On the bottom-left corner of each bounding box, we display the surface (stuff) property of the two bottom corners.

2. the supporting surface should be one of the following: ‘water’, ‘ground’ (exception: ‘platform’), ‘solid’, ‘vegetation’: (exception: ‘flower’, ‘tree’), ‘floor’, ‘plant/grass’, ‘plant/flower’, ‘structural/net’. For cars in the multi-category setting, they are ‘ground’(exception: ‘platform’), ‘solid’, ‘vegetation’ (exception: ‘flower’, ‘tree’), ‘floor’, ‘plant/grass’, ‘plant/flower’, ‘wall/wall-concrete’, ‘building/building-other’. The supporting surface is obtained by looking at both bottom corners of the person or object bounding box. At least one of them should be in the previous categories;
3. for people, we look for amodal observations, where both the head and ankles must be visible—since our model assumes full object visibility and that each bounding box be lying on the ground, similar to [3];
4. for people, the occlusion should be minimal. The person or object of interest should occupy the largest region in the bounding box, otherwise is it discarded.

Data Examples More data examples from COCO-Scale can be found in Fig. 3. Despite our best efforts, some annotations breaking our assumptions pass our data filtering. For example, in the second column, second row, the person

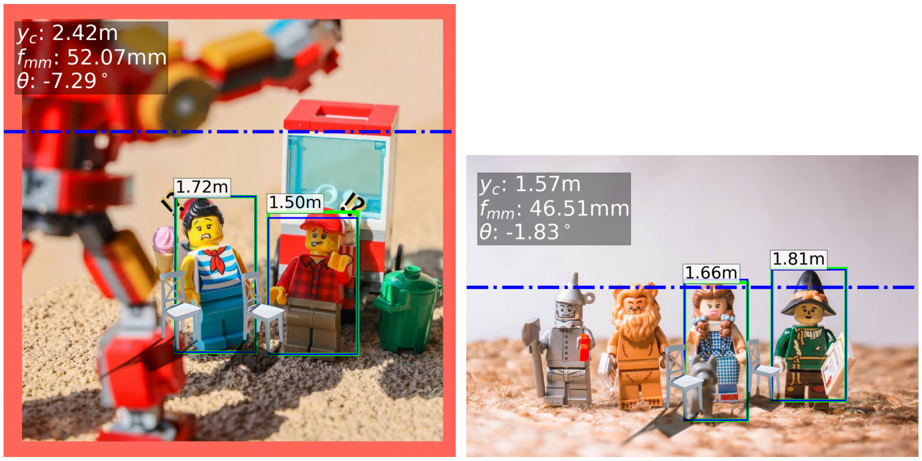

Fig. 2: Demo of two real-world samples where virtual object insertion is performed with our method.

standing on top of the platform has a valid supporting surface ('floor-stone'), despite not lying on the dominant ground plane.

5.2 KITTI Dataset

Sample data from the KITTI dataset can be found in Fig. 6. In Fig. 4, we provide histograms of both estimated and ground truth heights, showing the distribution of our estimations matches the ground truth.

5.3 IMDB-23K Dataset

Results on the IMDB-23K dataset are shown in Fig. 11. In Fig. 5, we show the distribution of estimated and ground truth people height.

6 More Results

6.1 Failure Cases

Failure cases of our method can be found in Fig. 9. Image 1, 2, 5, 6 demonstrate cases of large camera roll which our camera model cannot deal with. Image 1, 3, 5 shows failure due to extreme camera height which exceeds our camera height classification range. Image 1, 2, 4, 5, 6 show wrong horizon estimation. In particular image 6 shows image distortion which violates our perspective camera model, and image 4 includes outlier objects that do not situate on the ground.

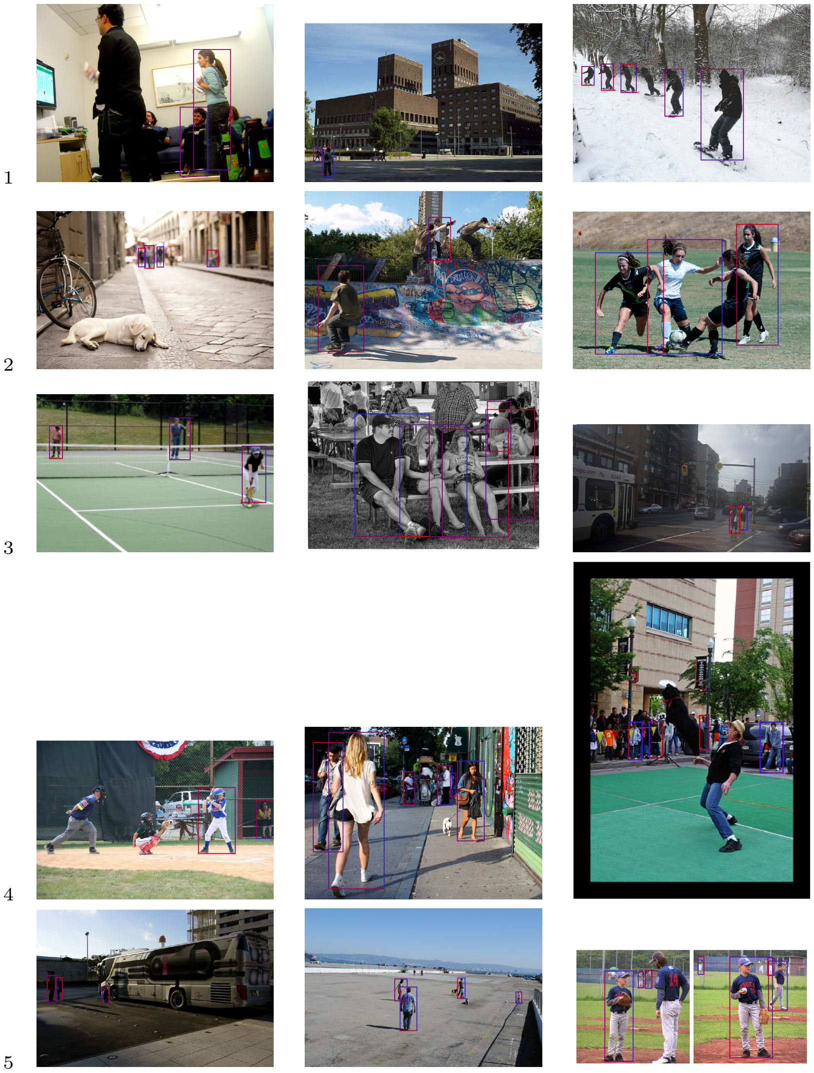

Fig. 3: Examples of COCO-Scale-person. Each bounding box is a valid person acquired from our pruning.

Real world cases of scale ambiguity In Fig. 2 we show two demo cases of Lego figure macro shots where figures are detected and our method is performed for scene parameters estimation and virtual object insertion. We demonstrate that although domain gap exists between our training data and test images, we are able to successfully detect reference objects and apply our method to those images which satisfy our scene model. However the issue is, we are not able to hand wave away the inherent scale ambiguity between real person and Lego

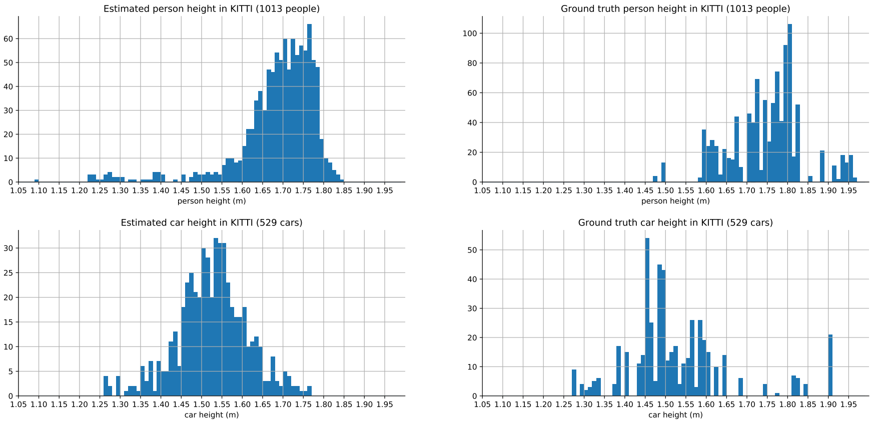

Fig. 4: (Top) Histogram of estimated heights and ground truth heights of all 'person' objects in the KITTI dataset. (Bottom) Histogram of estimated heights and ground truth heights of all 'car' objects in the KITTI dataset

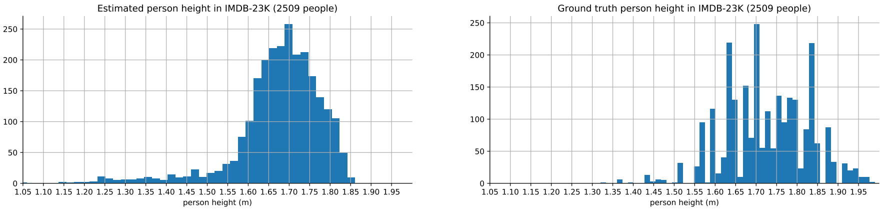

Fig. 5: Histogram of estimated heights and ground truth heights of all 'person' objects in the IMDB-23K dataset

figures (similar to the demo in [1]) because our person height prior is based on real person thus does work on mini-sized Lego figures.

6.2 Corner Cases

Corner cases can be found in Fig. 10. We show in row 1 & 2 cases where there are people of abnormal height (*e.g.* kids or people sitting) or people that are close to the camera (which violates the approximated camera model in PGM), in row 3 cases where there are extreme camera heights other than street-level camera height, in row 4 & 5 cases where horizons are away from the image center or are incorrectly estimated, in row 6 cases where there are outlier objects which are not situated on the ground. We demonstrate that our proposed method performs better than the baseline model in most of these cases.

6.3 More results on COCO-Scale

More results on COCO-Scale can be found in Fig. 7 and Fig. 8.

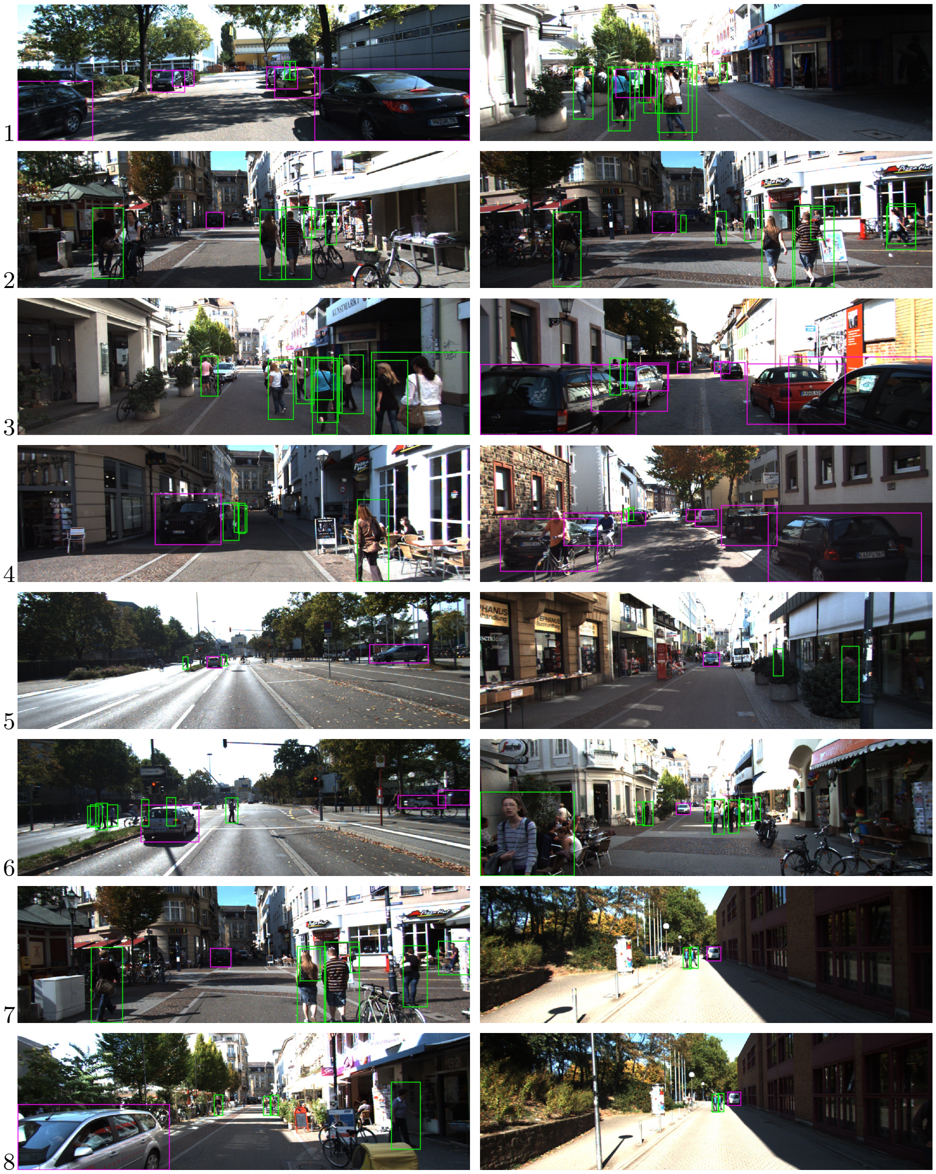

Fig. 6: Examples of KITTI. Each bounding box is a valid object acquired from our pruning (pedestrians in green and cars in magenta).

6.4 More results on IMDB-23K

More results on IMDB-23K are shown in Fig. 11. Comparison between our model which estimates the keypoints and models the upright ratio (SN-L3-kps-can) and our vanilla model (SN-L3) can be found in Fig. 12. The comparison shows, by

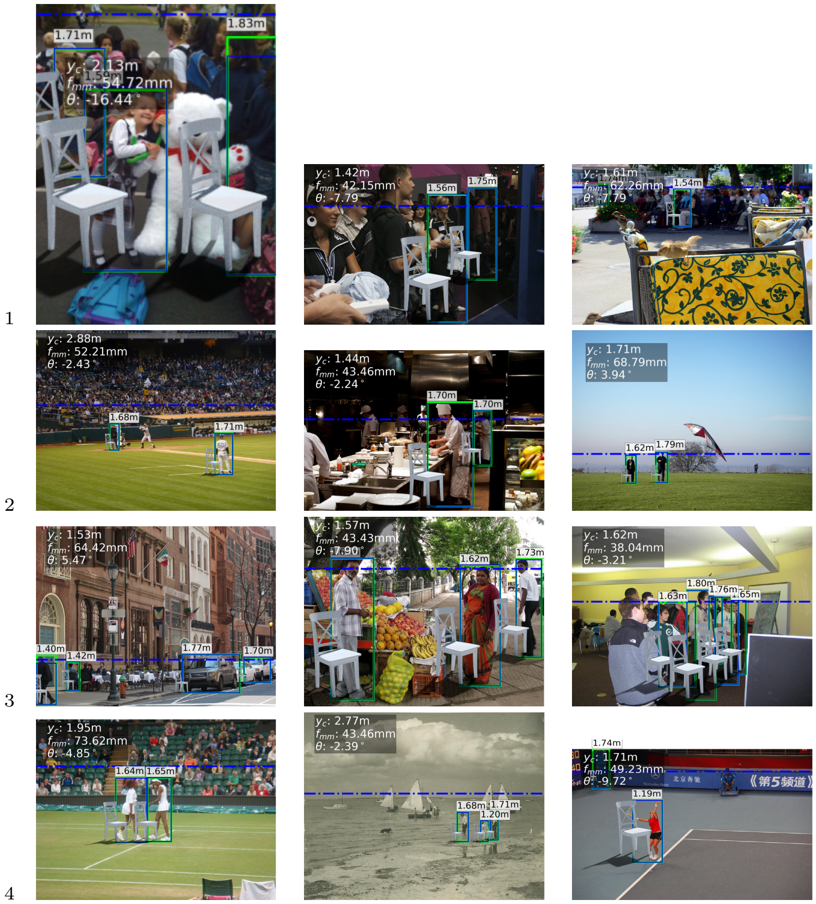

Fig. 7: More results of scene parameters estimation and virtual object insertion on COCO-Scale with model SN-L3-mulCat. The detected boxes are shown in green and reprojected ones in blue. The horizon is shown as a dashed blue line. Camera parameters are overlaid on the top. Chairs of 1m height are inserted alongside each person with the estimated parameters.

incorporating keypoint prediction and upright ratio discount, that SN-L3-kps-can is better able to deal with situations of person that is not standing upright, extending Table 6 of the main paper. In Fig. 5 we give a histogram of the estimated height as well as ground truth height of person and car.

6.5 More results on KITTI

More results on KITTI-mulCat of our best performing model (SN-L3-mulCat) can be found in Fig. 14. We also give a histogram of the estimated height as well as ground truth height of person and car in Fig. 4.

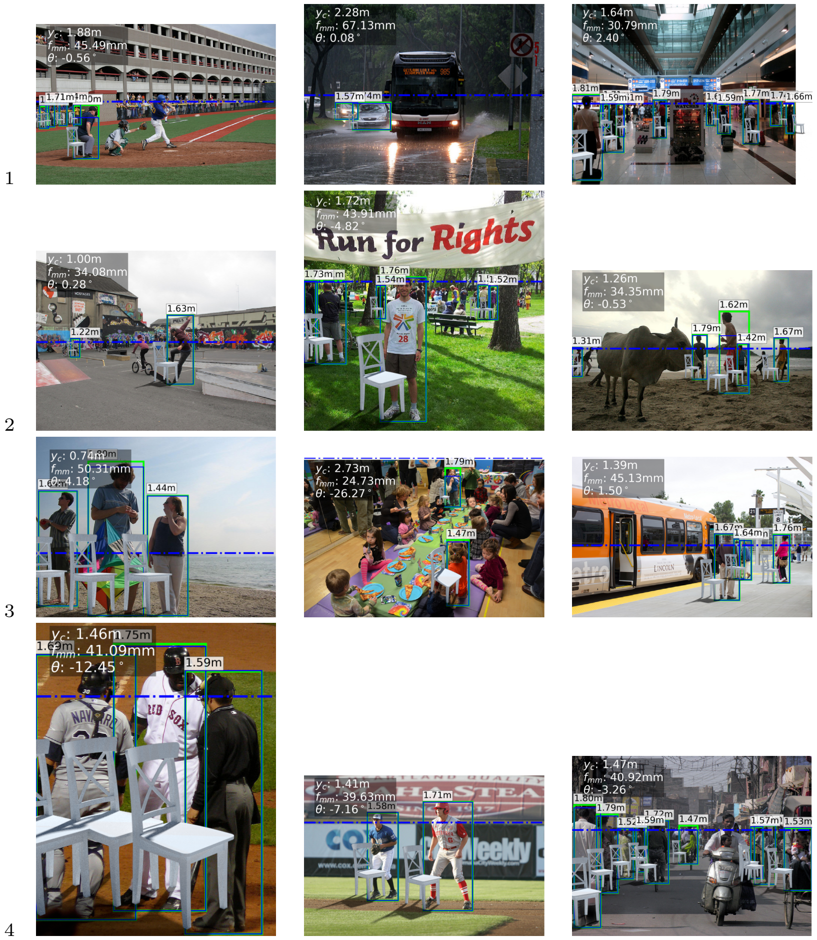

Fig. 8: More results of scene parameters estimation and virtual object insertion on COCO-Scale with model SN-L3-mulCat (cont.)

Comparison by with multiple category as input in Fig. 14 we give a comparison for model SN-L3-mulCat where only pedestrians are used as input into the model, against both pedestrians and cars are input. We are able to demonstrate, by using extra objects from the other category, we are able to get better person height estimation (row 1, 3, 4, 6) or better camera heights (row 1, 2, 3, 4, 5) because they are better constrained when larger number of objects from multiple categories are used.

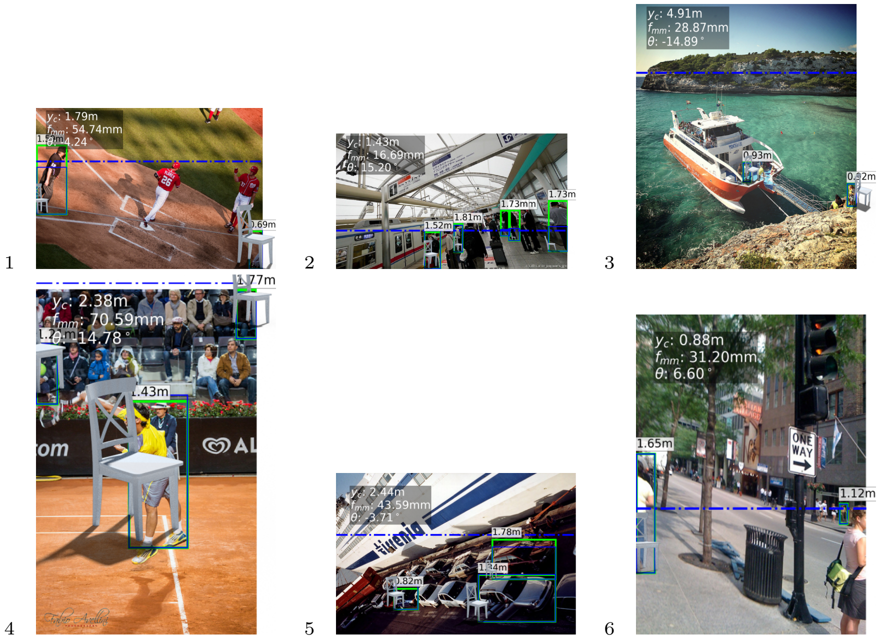

Fig. 9: Failure cases results of scene parameters estimation and virtual object insertion on COCO-Scale with model COCO-Scale-mulCat.

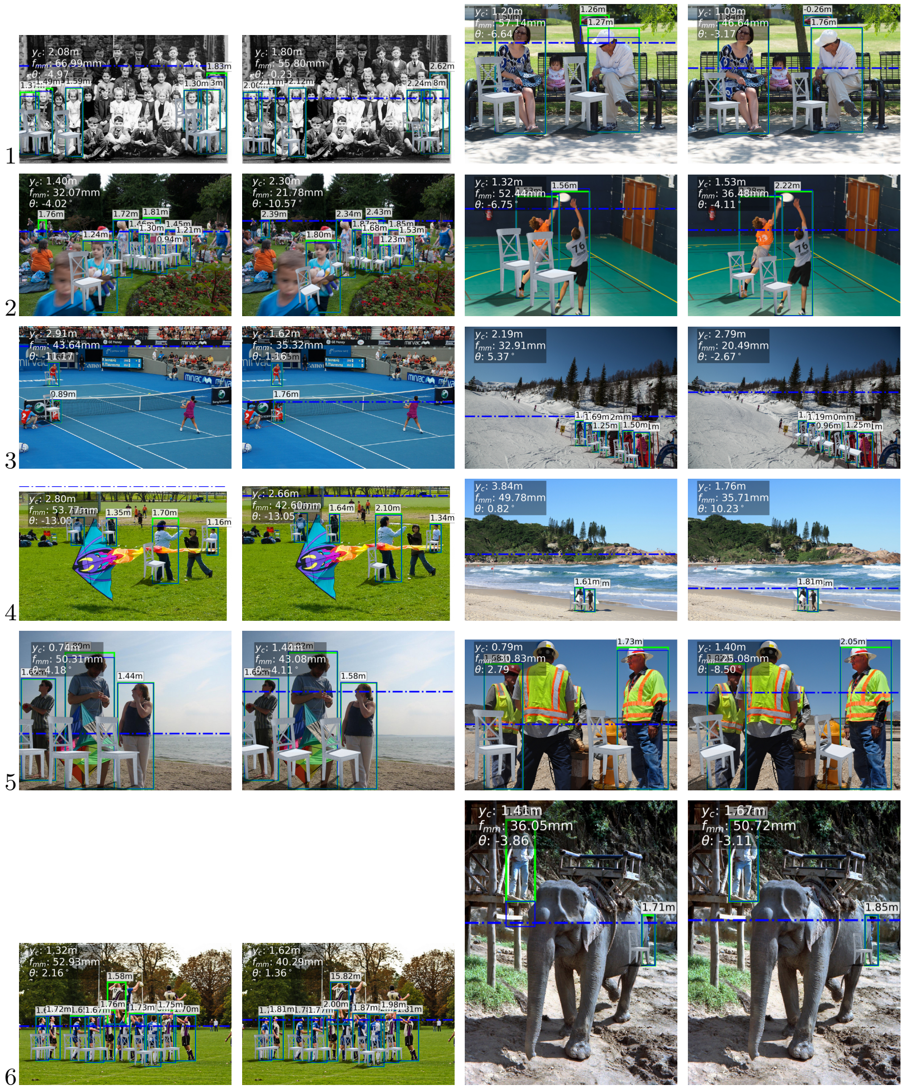
1
y_c: 2.08m f_{mm}: 66.99mm θ: -4.97
y_c: 1.80m f_{mm}: 55.80mm θ: -0.23
y_c: 1.20m f_{mm}: 57.14mm θ: -6.64
y_c: 1.09m f_{mm}: 46.64mm θ: -3.17
2
y_c: 1.40m f_{mm}: 32.07mm θ: -4.02°
y_c: 2.30m f_{mm}: 21.78mm θ: -10.57°
y_c: 1.32m f_{mm}: 52.44mm θ: -6.75°
y_c: 1.53m f_{mm}: 36.48mm θ: -4.11°
3
y_c: 2.91m f_{mm}: 43.64mm θ: -11.17
y_c: 1.62m f_{mm}: 35.32mm θ: 1.16
y_c: 2.19m f_{mm}: 32.91mm θ: 5.37°
y_c: 2.79m f_{mm}: 20.49mm θ: -2.67°
4
y_c: 2.80m f_{mm}: 53.77mm θ: -13.08
y_c: 2.66m f_{mm}: 42.60mm θ: -13.05
y_c: 3.84m f_{mm}: 49.78mm θ: 0.82°
y_c: 1.76m f_{mm}: 35.71mm θ: 10.23°
5
y_c: 0.74m f_{mm}: 50.31mm θ: 4.18°
y_c: 1.44m f_{mm}: 43.08mm θ: 4.11°
y_c: 0.79m f_{mm}: 30.83mm θ: 2.79
y_c: 1.40m f_{mm}: 25.08mm θ: -8.50
6
y_c: 1.32m f_{mm}: 52.93mm θ: 2.16°
y_c: 1.62m f_{mm}: 40.29mm θ: 1.36°
y_c: 1.41m f_{mm}: 36.05mm θ: -3.86
y_c: 1.67m f_{mm}: 50.72mm θ: -3.11

Fig. 10: Comparison on corner cases between SN-L3-mulCat (left in each pair), and PGM (right in each pair) on COCO-Scale-mulCat.

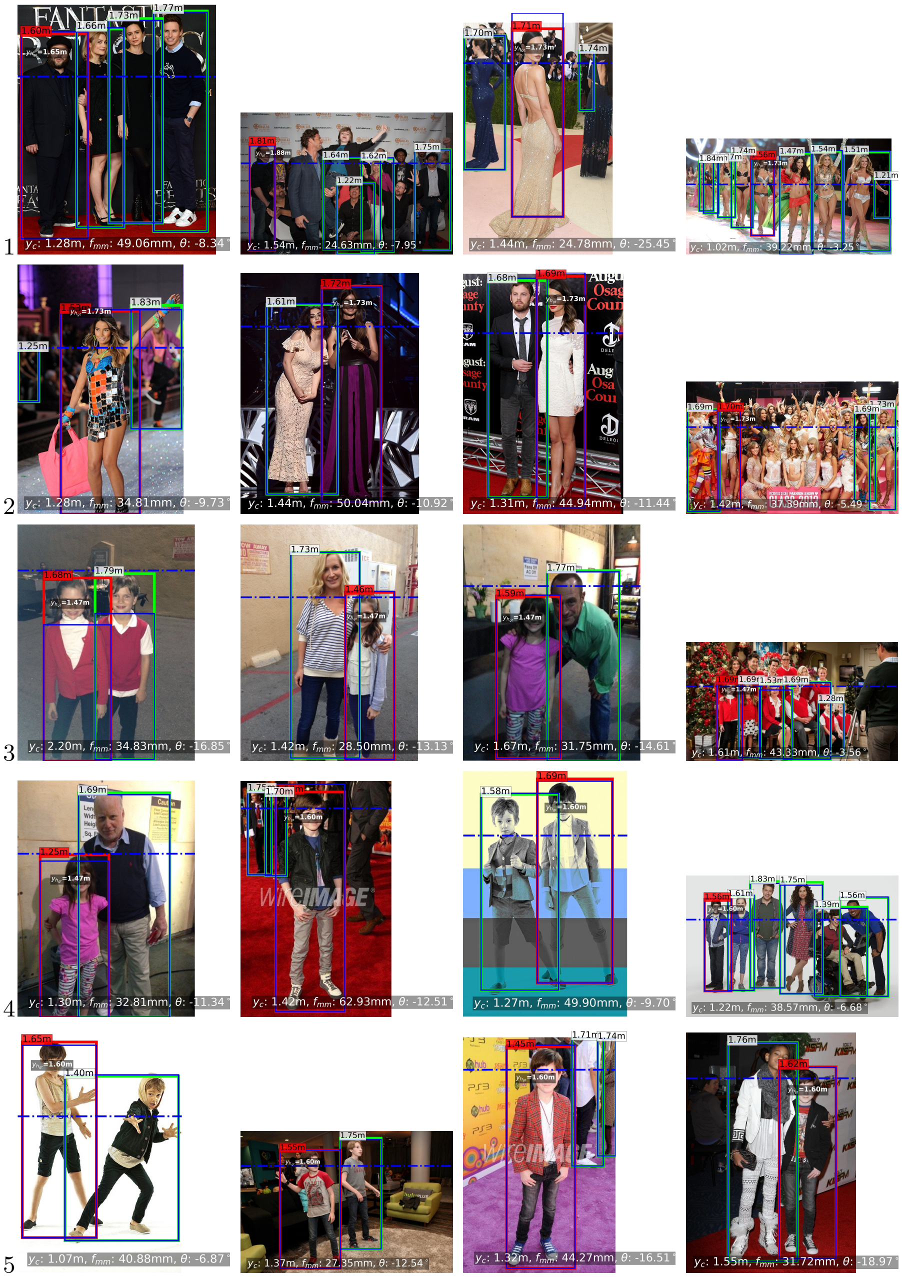
1
1.60m
1.66m
1.73m
1.77m
y_{c}: 1.28m, f_{mm}: 49.06mm, θ: -8.34°
1.81m
1.64m
1.62m
1.75m
1.22m
y_{c}: 1.54m, f_{mm}: 24.63mm, θ: -7.95°
1.70m
1.71m
1.74m
y_{c}: 1.44m, f_{mm}: 24.78mm, θ: -25.45°
1.84m
1.74m
1.47m
1.54m
1.51m
1.21m
y_{c}: 1.02m, f_{mm}: 39.22mm, θ: -3.25°
2
1.25m
1.83m
y_{c}: 1.28m, f_{mm}: 34.81mm, θ: -9.73°
1.61m
1.72m
y_{c}: 1.44m, f_{mm}: 50.04mm, θ: -10.92°
1.68m
1.69m
y_{c}: 1.31m, f_{mm}: 44.94mm, θ: -11.44°
1.69m
1.70m
1.69m
y_{c}: 1.42m, f_{mm}: 37.39mm, θ: -5.49°
3
1.68m
1.79m
y_{c}: 2.20m, f_{mm}: 34.83mm, θ: -16.85°
1.73m
1.46m
y_{c}: 1.42m, f_{mm}: 28.50mm, θ: -13.13°
1.59m
1.77m
y_{c}: 1.67m, f_{mm}: 31.75mm, θ: -14.61°
1.69m
1.53m
1.69m
1.28m
y_{c}: 1.61m, f_{mm}: 43.33mm, θ: -3.56°
4
1.69m
1.25m
y_{c}: 1.30m, f_{mm}: 32.81mm, θ: -11.34°
1.70m
y_{c}: 1.42m, f_{mm}: 62.93mm, θ: -12.51°
1.58m
1.69m
y_{c}: 1.27m, f_{mm}: 49.90mm, θ: -9.70°
1.83m
1.75m
1.61m
1.39m
1.56m
y_{c}: 1.22m, f_{mm}: 38.57mm, θ: -6.68°
5
1.65m
1.40m
y_{c}: 1.07m, f_{mm}: 40.88mm, θ: -6.87°
1.55m
1.75m
y_{c}: 1.37m, f_{mm}: 27.35mm, θ: -12.54°
1.45m
1.71m
1.74m
y_{c}: 1.32m, f_{mm}: 44.27mm, θ: -16.51°
1.76m
1.62m
y_{c}: 1.55m, f_{mm}: 31.72mm, θ: -18.97°

Fig. 11: More results of scene parameters estimation with SN-L3-kps-can on IMDB-23K. Person in red bounding box is the annotated person. The estimated height and ground truth height are labelled on the upper-left corner and over the bounding box respectively.

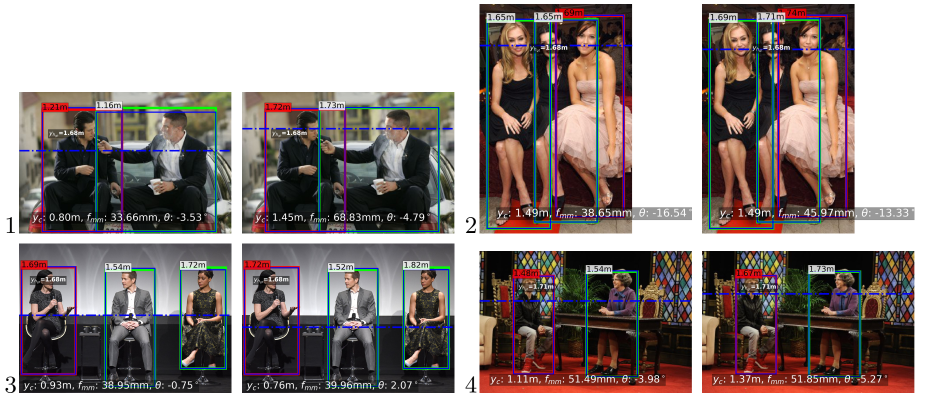
1
y_c: 0.80m, f_{mm}: 33.66mm, θ: -3.53°
y_c: 1.45m, f_{mm}: 68.83mm, θ: -4.79°
2
y_c: 1.49m, f_{mm}: 38.65mm, θ: -16.54°
y_c: 1.49m, f_{mm}: 45.97mm, θ: -13.33°
3
y_c: 0.93m, f_{mm}: 38.95mm, θ: -0.75°
y_c: 0.76m, f_{mm}: 39.96mm, θ: 2.07°
4
y_c: 1.11m, f_{mm}: 51.49mm, θ: -3.98°
y_c: 1.37m, f_{mm}: 51.85mm, θ: -5.27°

Fig. 12: Comparison between SN-L3-kps-can which incorporates keypoint estimation and upright ratio estimation (left), and SN-L3 which does not (right).

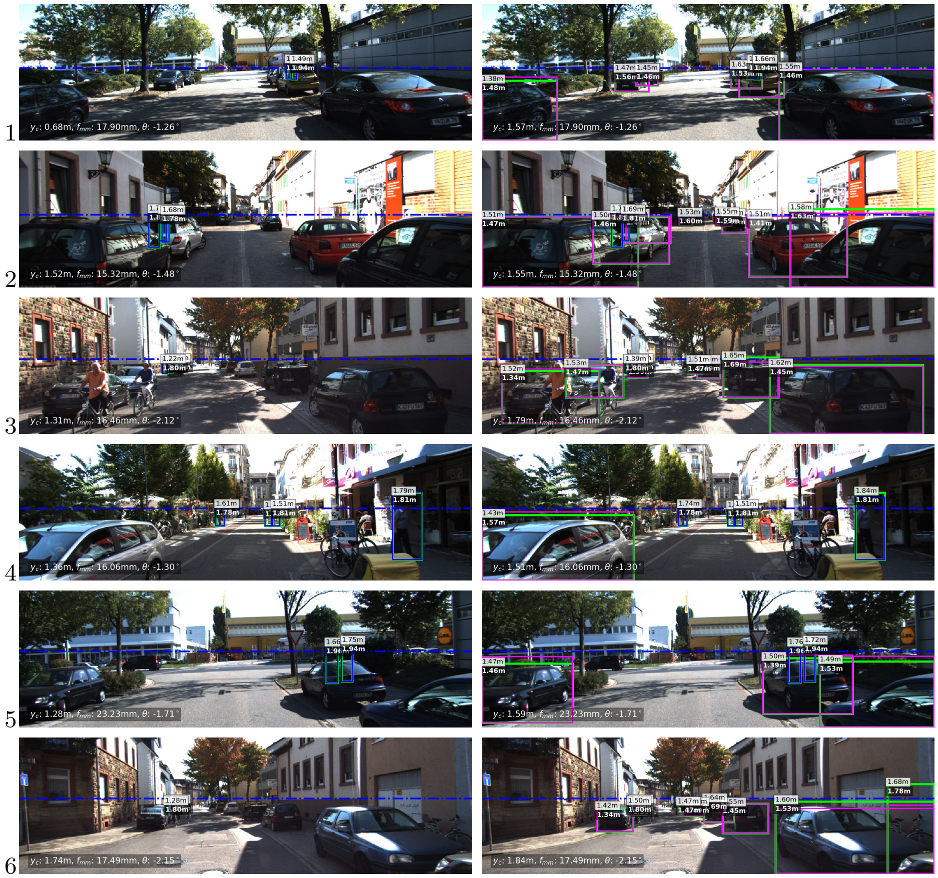
1
y_c: 0.68m, f_{mm}: 17.90mm, θ: -1.26°
y_c: 1.57m, f_{mm}: 17.90mm, θ: -1.26°
2
y_c: 1.52m, f_{mm}: 15.32mm, θ: -1.48°
y_c: 1.55m, f_{mm}: 15.32mm, θ: -1.48°
3
y_c: 1.31m, f_{mm}: 16.46mm, θ: -2.12°
y_c: 1.79m, f_{mm}: 16.46mm, θ: -2.12°
4
y_c: 1.36m, f_{mm}: 16.06mm, θ: -1.30°
y_c: 1.51m, f_{mm}: 16.06mm, θ: -1.30°
5
y_c: 1.28m, f_{mm}: 23.23mm, θ: -1.71°
y_c: 1.59m, f_{mm}: 23.23mm, θ: -1.71°
6
y_c: 1.74m, f_{mm}: 17.49mm, θ: -2.15°
y_c: 1.84m, f_{mm}: 17.49mm, θ: -2.15°

Fig. 13: Comparison between SN-L3-mulCat with person only input (left), and SN-L3-mulCat with both person and car as input.

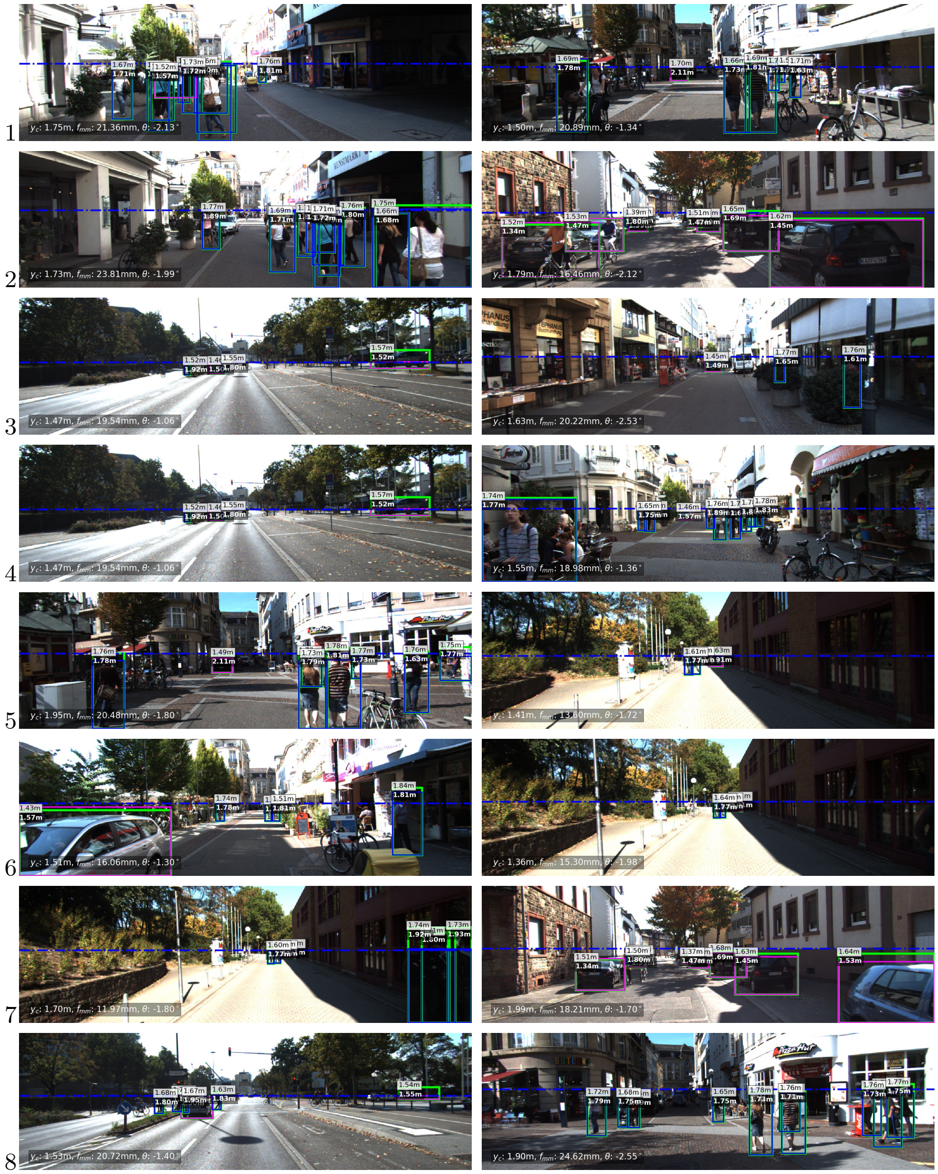
1
y_c: 1.75m, f_{mm}: 21.36mm, θ: -2.13°
y_c: 1.50m, f_{mm}: 20.89mm, θ: -1.34°
2
y_c: 1.73m, f_{mm}: 23.81mm, θ: -1.99°
y_c: 1.79m, f_{mm}: 16.46mm, θ: -2.12°
3
y_c: 1.47m, f_{mm}: 19.54mm, θ: -1.06°
y_c: 1.63m, f_{mm}: 20.22mm, θ: -2.53°
4
y_c: 1.47m, f_{mm}: 19.54mm, θ: -1.06°
y_c: 1.55m, f_{mm}: 18.98mm, θ: -1.36°
5
y_c: 1.95m, f_{mm}: 20.48mm, θ: -1.80°
y_c: 1.41m, f_{mm}: 13.60mm, θ: -1.72°
6
y_c: 1.51m, f_{mm}: 16.06mm, θ: -1.30°
y_c: 1.36m, f_{mm}: 15.30mm, θ: -1.98°
7
y_c: 1.70m, f_{mm}: 11.97mm, θ: -1.80°
y_c: 1.99m, f_{mm}: 18.21mm, θ: -1.70°
8
y_c: 1.53m, f_{mm}: 20.72mm, θ: -1.40°
y_c: 1.90m, f_{mm}: 24.62mm, θ: -2.55°

Fig. 14: More results of Scene parameters estimation with SN-L3-mulCat on KITTI-mulCat. Reprojected pedestrians are in blue, while cars are in magenta.

References

1. Ham, C., Lucey, S., Singh, S.: Hand waving away scale. In: European conference on computer vision. pp. 279–293. Springer (2014)
2. Hold-Geoffroy, Y., Sunkavalli, K., Eisenmann, J., Fisher, M., Gambaretto, E., Hadap, S., Lalonde, J.F.: A perceptual measure for deep single image camera calibration. In: CVPR. pp. 2354–2363 (2018)
3. Kar, A., Tulsiani, S., Carreira, J., Malik, J.: Amodal completion and size constancy in natural scenes. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 127–135 (2015)
4. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
5. Massa, F., Girshick, R.: maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch. https://github.com/facebookresearch/maskrcnn-benchmark (2018), accessed: [Oct 16, 2019]