Supplemental Materials: Objectron: A Large Scale Dataset of Object-Centric Videos in the Wild with Pose Annotations

Appendix A. Related Work Comparison

Table 1 provides a comparison between Objectron and similar datasets for 3D object detection and understanding.

Appendix B. Details of the Baseline 3D Object Detection Models

In section 5.2 we evaluated two baseline models, namely MobilePose and a Two-Stage architecture, Figure 1 shows the overview architecture of the models that we used as baselines. Both models are capable of achieving real-time performance on mobile devices.

The original MobilePose network also uses the shape information obtained from synthetic data. However, we showed it also works by training purely on real data without any shape information. In our implementation, we used MobileNetV2 as backend, and added two heads to the network: 1) An attention head that creates an attention mask at the center keypoint of the 3D bounding box, and 2) a regression head, that predicts the x-y adjustment of the eight other keypoint from the center keypoint. The network predicts the nine 2D projected keypoints, which are later lifted to 3D using EPNP algorithm [16].

The two-stage network first uses a 2D object detector (SSD network in our implementation) to detect a 224×224 crop of the object. Then the network (as shown in Figure 1) uses an EfficientNet-lite network to encode the input image to an $7 \times 7 \times 1152$ embedding vector, followed by a fully connected layer to regress the 9 2D keypoints. The network uses a similar EPnP algorithm as in [16] to lift the 2D predicted keypoints to 3D. The hyper-parameters of the training jobs are provided in Table 2.

Appendix C. Details of the Objectron data format

The data is stored in [link-redacted-for-peer-review] for public access. For each sample, the dataset provides the raw video file (in MOV file format, at 30fps, and 1440×1920 resolution, the AR Metadata, and the annotation result. The AR metadata contains the camera transformation, view, projection, and intrinsic matrix. The camera transformation contains the camera transformation from the first frame in the sequence. Furthermore, the sparse point-cloud in the world-coordinate and surface planes (including the normal and extend, boundary points, and plane alignment w.r.t. gravity vector) are provided.

For each object instance, the annotation data includes the bounding box's orientation, translation, and scale, as well as the 3D vertices in the world and camera coordinate system and their 2D projection (with depth) in the image plane. Each instance has a label string that corresponds to the object's category. The bounding box transformation transforms an axis-aligned unit bounding box to the annotated bounding box in the world-coordinate system. For each frame in the video, we also compute the transformed bounding box in the camera coordinate system as well.

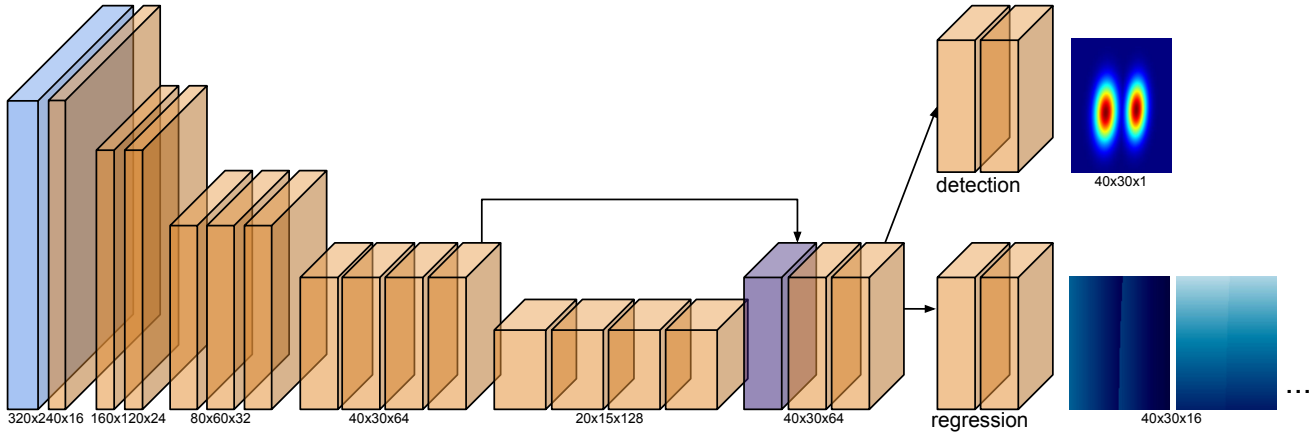
Besides raw data, we also provided a pre-processed dataset that can be easily connected to existing input pipelines for model training. We converted our entire dataset to Tensorflow tf.Example and tf.SequenceExample for image and video models, respectively. Both formats are stored as Tensorflow records. We implementing example pipelines to feed this data to PyTorch, Tensorflow, and Jax training pipelines efficiently.

Dataset	Size	# Categories	Annotation	Reconstruction	Multi-view	CAD
NYU v2 [26]	464 scans		2D LabelMe style		yes	some
Sun 3D [19]	415 Scans		2D Polygons	Aligned poses	yes	no
Sun RGB-D [29]	10k frames	800	3D annotation on 2D image	Dense 3D	no	no
ScanNet [6]	1513 videos 2.5M frames		dense 3D	Dense 3D	yes	yes
Pascal3D+ [39]	31k frames	12	2D-3D alignment	No	no	yes
ObjectNet3D [38]	90k frames	100	2D-3D alignment	No	no	yes
IKEA [18]	759 frames	11	2D-3D alignment	No	no	yes
3DObject [30]	6675 images	10	discretized view	No	no	no
RIO [35]	1482 frames		3D object alignment	Dense 3D	yes	no
Objectron	15K videos 4M frames	9	3D object pose	Neural 3D	yes	no

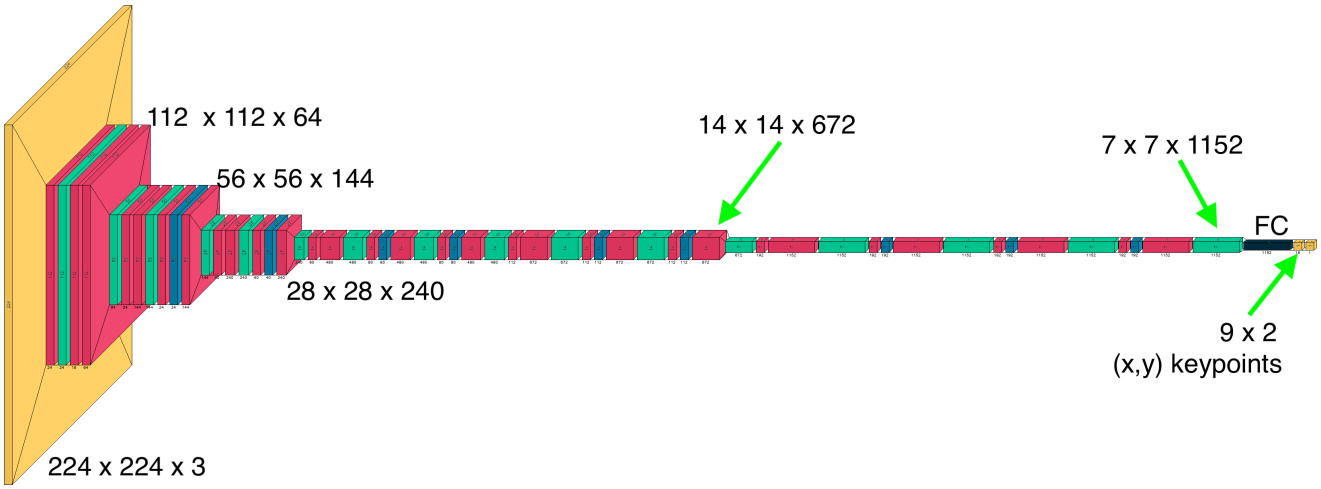
Table 1: Overview of datasets for 3D Object detection and understanding.

	MobilePose	Two-stage
Epoch	100	250
Learning rate	1e-2, decayed to 1e-3 in 30 epoch	1e-2 to 1e-6 exponential annealing
Batch-size	128	64
Optimizer	adam	adam
Input-size	$480 \times 640 \times 3$	$224 \times 224 \times 3$
Output-size	120×160	9×2
Loss	MSE on attention + L_1 on regression	Per vertex MSE normalized on diagonal edge length
Backend	MobileNetV2	EfficientNet-Lite

Table 2: Hyper parameters for the baseline models.



(a) The architecture of the MobilePose[16] model.



(b) Architecture of the two-stage model. The red blocks are 1×1 convolutional layers, green blocks are depthwise convolutional layers, and blue blocks are addition layers for skip connection. The black block at the end is a fully connected layer.

Figure 1: The baseline models used for 3D Object Detection task.