

## 6 SUPPLEMENTARY MATERIAL

### 6.1 CODE

The anonymised code repository can be found here<sup>2</sup>, which also includes links to download the CLEVR-MRT dataset and visualise it. Note that for the sake of anonymity, some scripts in the codebase were omitted.

### 6.2 HYPERPARAMETERS

The table below lists hyperparameters of the FILM module, which can be found in `architectures/clevr/probe.py`.<sup>3</sup>

- `n_in`: number of input feature maps (this is 1024 for the ImageNet pre-trained encoder since it outputs a feature stack of dimension  $(1024, 14, 14)$ , but in the 3D case it is 128)
- `rnn_dim`: output RNN embedding dimensionality
- `rnn_num_layers`: number of hidden layers in the RNN
- `n_resblocks`: how many FILMed ResBlocks do we use (this was always set to 4)
- `with_coords`: append 2D coordinates (Liu et al., 2018) in each FILMed ResBlock?
- `coord_shape`: just a nuisance hyperparameter to define the spatial dimension of the coord conv feature maps that are concatenated inside the resblock
- `nf`: number of feature maps in resblocks. If this is set to None, it will simply default to  $(rnn\_dim + ncf + ncf) / 2$ .
- `with_camera`: do we concatenate the camera embedding with the RNN embedding?
- `ncf`: the linear projection layer which maps camera (6 coords) to this embedding size (if `with_camera` is True). If this is None, it will simply be default to 6
- `weight_decay`: L2 weight decay
- `imagenet_scaling`: use ImageNet model’s mean/std to scale the input? (If not True, we shift and scale input by 0.5 and 0.5, so that it lies in the range  $[-1, +1]$ ).
- `is_3d`: we use 3D resblocks instead of 2D resblocks (set to true when training FILM on volumes)

Each experiment was trained for a maximum of 60 epochs with the ADAM optimiser (Kingma & Ba, 2014), with a default learning rate of  $3 \times 10^{-4}$  and first and second moment coefficients  $\{\beta_1, \beta_2\} = \{0.9, 0.999\}$ . Figure 6 illustrates an example range of hyperparameters explored per experiment (where each experiment refers to one of the results performed in Table 1). Note that this is *not* an exhaustive range of values explored – rather, the values seen per experiment in the Figure 6 correspond to the batch of runs in which at least one of the runs inside the batch gave the highest validation score(s). (Other batches of HP/value combinations were also run but may not have yielded the best validation accuracies, and therefore are not shown in the figure.) Whichever experiment was found to have the highest validation score was re-trained multiple times (under different seeds) and evaluated on the test set.

### 6.3 MAC BASELINES

Code for our MAC baseline was adapted from here<sup>4</sup>. In short, a bidirectional LSTM was used here with 12 time steps used for the MAC reasoning step. In order to leverage camera information, we simply concatenated the camera’s embedding to the summary question embedding (not to the contextual word embeddings). For the best performing experiment, self-attention was disabled.

<sup>2</sup><https://github.com/anonymousscat2434/clevr-mrt>

<sup>3</sup>While an effort was made for this information to be accurate, the source code should always be the definitive reference.

<sup>4</sup><https://github.com/rosinality/mac-network-pytorch.git>

```

2d film, no camera
-----
{'rnn_dim': {128, 512, 256, 1024}, 'n_resblocks': {4}, 'encoder': {'gru'}, 'with_coords': {True}, '
  ↳ rnn_num_layers': {1, 2}, 'n_in': {1024}, 'nf': {None}, 'coord_shape': {(14, 14)}, 'with_camera': {
  ↳ False}, 'ncf': {None}}
{'weight_decay': {1e-05, 0.0001}}

2d film, camera
-----
{'rnn_dim': {1024, 512}, 'n_resblocks': {4}, 'encoder': {'gru'}, 'with_coords': {True}, 'rnn_num_layers': {1,
  ↳ 2}, 'n_in': {1024}, 'nf': {64, 128}, 'coord_shape': {(14, 14)}, 'with_camera': {True}, 'ncf': {64}}
{'weight_decay': {0, 0.0001, 1e-05}}
{'imagenet_scaling': {False}}
{'batch_size': {16}}

3d film, no camera embed/rotation
-----
{'rnn_dim': {512, 1024}, 'rnn_num_layers': {1}, 'n_resblocks': {4}, 'encoder': {'gru'}, 'with_coords': {True
  ↳ }, 'coord_shape': {(16, 14, 14)}, 'with_camera': {False}, 'ncf': {None}, 'flatten_3d': {False}, '
  ↳ is_3d': {True}, 'n_in': {128}, 'nf': {64, 128}}
{'weight_decay': {0, 1e-05}}
{'imagenet_scaling': {True}}

3d film, camera rotation
-----
{'rnn_dim': {512, 1024}, 'rnn_num_layers': {1, 2}, 'n_resblocks': {4}, 'encoder': {'gru'}, 'with_coords': {
  ↳ True}, 'coord_shape': {(16, 14, 14)}, 'with_camera': {False}, 'ncf': {None}, 'flatten_3d': {False},
  ↳ 'is_3d': {True}, 'n_in': {128}, 'nf': {64, 128}}
{'weight_decay': {0, 0.0001, 1e-05}}
{'imagenet_scaling': {True}}

3d film, camera embed
-----
{'rnn_dim': {512, 1024}, 'rnn_num_layers': {1, 2}, 'n_resblocks': {4}, 'encoder': {'gru'}, 'with_coords': {
  ↳ True}, 'coord_shape': {(16, 14, 14)}, 'with_camera': {True}, 'ncf': {None}, 'flatten_3d': {False}, '
  ↳ is_3d': {True}, 'n_in': {128}, 'nf': {64, 128}}
{'weight_decay': {1e-05, 0, 1e-06}}
{'imagenet_scaling': {True}}

3d film, both camera embed/rotation
-----
{'rnn_dim': {512, 1024}, 'rnn_num_layers': {1}, 'n_resblocks': {4}, 'encoder': {'gru'}, 'with_coords': {True
  ↳ }, 'coord_shape': {(16, 14, 14)}, 'with_camera': {True}, 'ncf': {64}, 'flatten_3d': {False}, 'is_3d
  ↳ ': {True}, 'n_in': {128}, 'nf': {64, 128}}
{'weight_decay': {0, 1e-05, 1e-06}}
{'imagenet_scaling': {True}}

```

Figure 6: The range of hyperparameters explored per experiment in Table 1 (the upper bound canonical baseline and contrastive experiments are not shown here).

Due to significant time constraints, hyperparameter tuning for this baseline was minimal. However, we can see in Table 1 that it obtains similar validation accuracy to the 2D FILM + camera conditioning baseline, and in principle can probably also be adapted to perform 3D reasoning.

#### 6.4 EXAMPLE IMAGES FROM DATASET

See Figure 7 and 8.

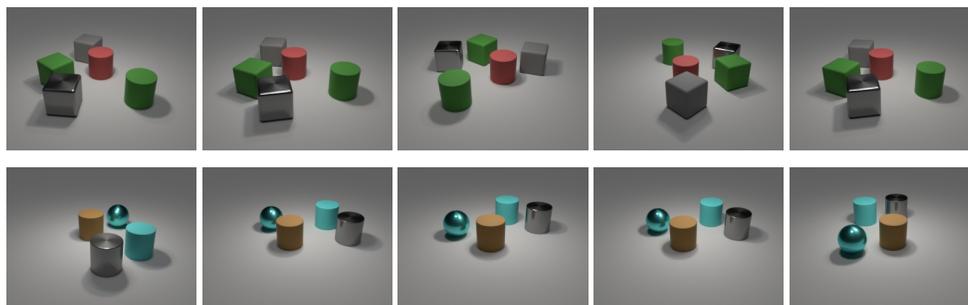


Figure 7: Random views of an example scene in CLEVR-MRT. The center image is the ‘canonical’ view.

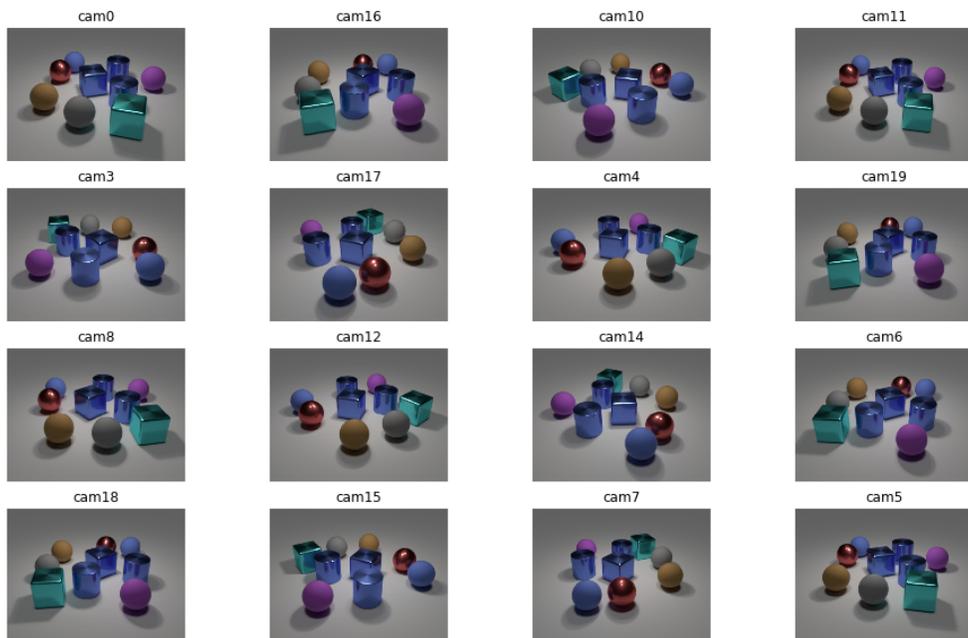


Figure 8: A full example of a CLEVR-MRT scene, showing 16 randomly sampled views of the scene (out of 20 in total). The canonical view is not shown here, but a sample of questions pertaining to the canonical view are:

- Q:** Are there the same number of large gray objects that are left of the gray thing and big matte cylinders? (**True**)
- Q:** Is the number of large brown things that are behind the red thing less than the number of purple objects? (**True**)
- Q:** Are there more blue blocks to the left of the large gray thing than blue matte spheres? (**False**)
- Q:** Are there an equal number of big brown things that are in front of the large brown rubber ball and big metal things that are behind the purple sphere? (**False**)