In this supplementary material we include (1) MUXNet hyperparameter search space in Section 1, (2) computational complexity of MUXNet and comparison to a combination of $1 \times 1 + 3 \times 3$ in Section 2, (3) effectiveness of MUXNet as a backbone semantic segmentation in Section 3.1, and (4) evaluation of generalization and robustness properties of MUXNet in Section 3.2. Finally Fig. 7 shows some qualitative object detection results on PASCAL VOC 2007, and Fig. 8 shows gradCam results on the ChestX-Ray14 dataset.

### 1. Search Space

![Figure 1: Search Space Encoding](image)

To encode the hyperparameter settings for a model, we first divide the model architectures into four stages, based on the spatial resolution of each layer’s output feature map. In each stage spatial resolution does not change. The first layer in each stage reduces the feature map size by half. For each stage, we search for kernel size (K) and expansion ratio (E). In addition, from second layer in each stage, we search for # of repetitions (N), # of input channels to compute convolution (G), leave-out ratio in channel multiplexing (L) and the spatial multiplexing setting (S) (see Fig. 1).

Table 1 summarizes the hyperparameters and available options for each stage. The obtained hyperparameters for our MUXNets are visualized in Figure 2. The total volume of the search space is approximately $14^{12}$.

### 2. Computational Complexity

In this section, we analytically compare the computational complexity of our MUXConv block (Figure 3b) with the widely-used MobileNet block [6]. For simplicity, we ignore the computation induced by the normalization and activation layers and we assume that for both blocks the number of input and output channels is the same i.e., $C$ channels.

![Figure 3: The visualization of the Mobilenet block (a) and our MUXConv block (b).](image)
Another evolution to expand, followed by a group-wise convolution \[9\] and convolution. The resulting number of parameters and the floating

\[\hat{C} = (1 - L) \cdot C\]

\[\text{Params} = \frac{C \cdot EC}{1 \times 1 \text{ conv}} + \frac{EC \cdot 3 \cdot 3}{3 \times 3 \text{ d.w. conv}} + \frac{EC \cdot C}{1 \times 1 \text{ conv}}\]

\[\text{FLOPs} = H \cdot W \cdot \left(\frac{C \cdot EC}{1 \times 1 \text{ conv}} + \frac{EC \cdot 3 \cdot 3}{3 \times 3 \text{ d.w. conv}} + \frac{EC \cdot C}{1 \times 1 \text{ conv}}\right)\]
We further evaluate the effectiveness of our models as backbones for the task of mobile semantic segmentation. We compare MUXNet-m with both MobileNetV2 [6] and ResNet18 [2] on ADE20K [12] benchmark. Additionally, we also compare two different segmentation heads. The first one, referred as C1, only uses one convolution module. And the other one, Pyramid Pooling Module (PPM), was proposed in [10]. All models are trained under the same setup: we use SGD optimizer with initial learning rate 0.02, momentum 0.9, weight decay 1e-4 for 20 epochs. Table 2 reports the mean IoU (mIoU) and pixel accuracy on the ADE20K validation set. MUXNet-m performs comparably with MobileNetV2 when paired with PPM, while being 1.5× more efficient in MAdds. We also provide qualitative visualization of semantic segmentation examples in Figure 5.

3.2. Generalization and Robustness

To further evaluate the generalization performance of our proposed models, we compare on a recently proposed benchmark dataset, ImageNetV2 [5], complementary to the original ImageNet 2012. We use the MatchedFrequency version of the ImageNet-V2. Figure 6a reports the top-5 accuracy comparison between our MUXNets and a wide-range of previous models. Even though there is a significant accuracy drop of 8% to 10% on ImageNet-V2 across models, the relative rank-order of accuracy on the original ImageNet validation set translates well to the new ImageNet-V2. And our MUXNet performs competitively on ImageNet-V2 as compared to other mobile models, such as ShuffleNetV2 [4], MobileNetV2 [6] and MnasNet-A1 [7].

The vulnerability to small changes in query images has always been a concern for designing better models. Hendrycks and Dietterich [3] recently introduced a new dataset, ImageNet-C, by applying commonly observable corruptions (e.g., noise, weather, compression, etc.) to the clean images from the original ImageNet dataset. The new dataset contains images perturbed by 19 different types of corruption at five different levels of severity. And we leverage this dataset to evaluate the robustness of our proposed models. Figure 6b compares Top-5 accuracy between our MUXNet-m and four other representative models, designed both manual and automatically. MUXNet-m performs favourably on ImageNet-C, achieving better accuracy on 18 out of 19 corruption types.

**References**

Figure 5: Examples from ADE20K validation set showing the ground truth (2nd row) and the scene parsing result (3rd row) from MUXNet-m. Color encoding of semantic categories is available from here.

Figure 6: (a) Generalization performance on ImageNet-V2 (MatchedFrequency) [5]. Numbers in the boxes indicate the drop in accuracy. (b) Robustness performance on ImageNet-C [3], which consist of ImageNet validation images corrupted by 19 commonly observable corruptions. Following the original paper that proposed ImageNet-C, we normalized the top-5 accuracy by AlexNet’s Top-5 accuracy. DARTS is from the author’s public Github repository. All other compared models are from Pytorch repository https://pytorch.org/docs/stable/torchvision/models.html.


[10] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In
Figure 7: Examples visualizing the detection performance of MUXNet-m on PASCAL VOC 2007 [1].

Atelectasis Cardiomegaly Effusion Infiltrate Pneumonia Pneumothorax

Figure 8: Examples of class activation map [11] of MUXNet-m on ChestX-Ray14 [8], highlighting the class-specific discriminative regions. The ground truth bounding boxes are plotted over the heatmaps.