# Supplementary Material for PatchDropout: Economizing Vision Transformers Using Patch Dropout

## A. Preprocessing

The images from CSAW are resized to $896 \times 896$, $448 \times 448$, $224 \times 224$ and $112 \times 112$ for the purposes of this work. The image resolution on IMAGENET varies and has an average resolution of $469 \times 387$. For PLACES365 we used images of size $256 \times 256$ and for CIFAR100 $32 \times 32$. We resize these to $448 \times 448$, $224 \times 224$, $128 \times 128$, $112 \times 112$ and $64 \times 64$ in our experiments using bi-linear interpolation.

## B. Models and training protocols

All models are initialized from IMAGENET-21K pre-trained weights and subsequently fine-tuned on the target task. Hyper-parameters are selected through grid search based on the result from the validation set. We use early stopping. The batch size is consistently 128 for all experiments, and each model is trained with an SGD optimizer with momentum set to 0.9. In every experiment, a linear learning rate warmup is utilized for the first 2 epochs. The learning rates are $3 \times 10^{-4}$ on IMAGENET, $5 \times 10^{-4}$ on CIFAR100 and PLACES365, and $10^{-4}$ on CSAW, following the results of our grid search. We use a weight decay of $1 \times 10^{-4}$ and label smoothing [25] of 0.1. We apply randomly resized cropping, random horizontal flipping, random rotation and color jittering as augmentations. Unless otherwise specified, each experiment was repeated three times and we report the mean. All the experiments are conducted with PyTorch [18]. The number of FLOPs is counted using the fvcore toolkit [21], and the allocated memory is calculated on a single GPU with a batch size of 128, unless otherwise stated.