A-ColViT : Real-time Interactive Colorization by Adaptive Vision Transformer

Anonymous submission

Abstract

Recently, the vision transformer (ViT) has achieved remarkable performance in computer vision tasks and has been actively utilized in colorization. Specifically, for point-interactive image colorization, previous research that uses convolutional layers is limited for colorizing partially an image, which produces inconsistent colors in an image. Thus, vision transformer has been used to alleviate this problem by using multi-head self attention to propagate user hints to distant relevant areas in the image. However, despite the success of vision transformers in colorizing the image and selectively colorizing the regions with user propagation hints, heavy underlying ViT architecture and the large number of required parameters hinder active real-time user interaction for colorization applications. Thus, in this work, we propose a novel efficient ViT architecture for real-time interactive colorization, A-ColViT that adaptively prunes the layers of vision transformer for every input sample. This method flexibly allocates computational resources of input samples, effectively achieving actual acceleration. In addition, we demonstrate through extensive experiments on ImageNet-ctest10k, Oxford 102flower, and CUB-200 datasets that our method outperforms the state-of-the-art approach and achieves actual acceleration.

Introduction

Despite the difficulty of colorization due to the requirement of a semantic understanding of the scenery and natural colors that dwell in the wild, various image colorization methods have shown remarkable results in restoring grayscale photographs as well as black and white films. In general, two colorization approaches contribute to the success of colorization: automatic colorization and user-guided colorization. First, as the name suggests, automatic colorization does not require any additional reference images or assistance. Previous automatic colorization methods generate *single colorized result*, yet colorization is an ill-posed problem with uncertainty (Zhang et al. 2017b). In contrast, user-guided colorization requires user-guided assistance or an image-based reference to narrow down suitable color distributions.

Among the user-guided colorization, the point-interactive colorization methods (Yin, Gong, and Qiu 2019; Levin, Lischinski, and Weiss 2004; Zhang et al. 2017b) help users with user-guided hints to assist in colorizing an image, while minimizing interaction with users. In particular, (Zhang et al. 2017b) proposed a colorization method with U-net architecture trained on ImageNet (Deng et al. 2009) and training with synthetically generated user hints through 2-D Gaussian sampling. However, prior works suffer from partial colorization, where the unclear boundary of images is not colored successfully. Furthermore, failure in consistent colorization comes from the difficulty of propagating hints to large and distant semantic regions. In order to tackle this problem, (Yun et al. 2022) leverage the architecture of vision transformers (ViT), allowing the model to learn to propagate the user hints to other distant and similar regions with self-attention. Also, they utilized local stabilizing layer for the effective upsampling process. Despite the exceptional performance of ViT in colorization applications, transformer-based models contain redundant computations resulting in slow inference speed. This problem limits users' active interactions on a variety of real-time colorization applications.

To address this challenging practical issue, we propose a novel flexible end-to-end framework A-ColViT, the real-time interactive colorization ViT that adaptively allocates computational resources of input images. Our proposed framework effectively utilizes a decision network to determine which redundant layer, such as the attention and Feed-Forward Network (FFN) layer, to skip in the transformer. In particular, we adapt Gumbel-Softmax trick (Maddison, Mnih, and Teh 2016) to enable backpropagation in the training process since the binary decisions from decision network are non-differentiable. In addition, we conduct extensive experiments on ImageNet-ctest10k, Oxford 102flowers, and CUB-200 datasets to validate the effectiveness of A-ColViT and demonstrate that our framework outperforms the state-of-the-art (SOTA) colorization methodology. Moreover, our visualization result illustrates whether computational resources are effectively allocated based on the easy and hard samples.

The main contributions of our work are summarized as follows:

- We propose A-ColViT, a flexible real-time user interactive colorization model, which input-adaptively allocates computational cost based on the easy and hard samples.
- We propose a trainable decision network that determines which redundant sub-layer of the transformer to skip or retain to achieve efficiency and real-time colorization

needs.

- Through extensive quantitative experiments and qualitative analysis, we demonstrate that our model outperforms the existing point-interactive colorization with vision transformer with improved inference speed.

Related Work

Interactive Colorization

Learning-based colorization methods do not require user interaction to generate adequate color images, while interactive methods require user-provided conditions to produce specified colored images. Reference-based colorization is one of the most popular interactive methods, which uses single reference images to provide overall color information (He et al. 2018; Bai et al. 2022; Zhang et al. 2017a). However, since the colorized image is highly dependent on a reference image, it is challenging for the user to modify particular regions in the colorized image. Moreover, the point-interactive colorization model enables users to provide precise $2 \times 2 \sim 7 \times 7$ color hints on particular input image regions to cover small regions of the full image, raising the importance of minimal user effort. Previous works detected simple patterns with image filters that determine the propagation portion of each hint, which is propagated within the region by optimization methods (Yin, Gong, and Qiu 2019; Levin, Lischinski, and Weiss 2004). Convolutional layers must be stacked deeply to propagate user hints from one region to a distant region, making the colorization of large semantic regions complex compared to relatively adjacent regions.

Colorization with Transformers

In contrast to the previous convolution-based technique in image synthesis, recent prior works utilized transformers (Kumar, Weissenborn, and Kalchbrenner 2021; Yin et al. 2021; Huang, Zhao, and Liao 2022; Ji et al. 2022) to automatically colorize images. (Kumar, Weissenborn, and Kalchbrenner 2021) proposed Colorization Transformer (ColTran) based on Axial Transformer (Ho et al. 2019) self-attention to unconditionally generate coarse low-resolution grayscale image and use color and spatial upsampler to produce high resolution colorized image. Also, hybrid transformer architectures are also proposed in colorization. (Ji et al. 2022) used transformer-based encoder and color memory decoder to obtain contextual semantics and color diversity, (Huang, Zhao, and Liao 2022) uses BERT-style hybrid transformer that utilizes input masked color tokens to restore the masked tokens via training on grayscale image. Also, (Yun et al. 2022) uses the Vision Transformer as a backbone and effectively upsampling the image through the local stabilizing layer. However, despite the superior performance of the transformer-based colorization method, it is still redundant for user-interactive applications.

Adaptive Inference

Pruning methods have demonstrated considerable performance in reducing model redundancy, while enhancing inference speed. In contrast to static pruning methods, recent adaptive inference methods adaptively perform pruning operations depending on input images. BlockDrop (Wu et al. 2018) and SkipNet (Wang et al. 2018) explored the dynamic method to skip blocks and layers, respectively. Several methods (Gao et al. 2018; Veit and Belongie 2018) were proposed to skip computations of unimportant channels. For example, FBS (Gao et al. 2018) dynamically amplifies and suppresses output channels. By skipping computations from unimportant channels, it is possible to use the previous layer's feature to predict the saliency of the output channel. Moreover, adaptive inference method has demonstrated acceleration in transformer-based models (Chen, Fan, and Panda 2021; Li et al. 2021; Wang et al. 2021; Yu et al. 2022; Yin et al. 2022; Meng et al. 2022). A-ViT (Yin et al. 2022) and DynamicViT (Rao et al. 2021) reduced the redundancy of the model by removing redundant image patches of each input, while AdaViT removed image patches, attention heads, and blocks. However, compared to reducing the redundancy of the layer, reducing the redundancy of image patches and attention heads possesses the overhead of inefficient path indexing and weight-copying. These restrain the actual acceleration. For real-time interactive colorization, we dynamically remove the unimportant attention and MLP layer in the transformer, and further improve the inference speed, compared to the previous transformer-based colorization approaches.

Method

In this work, we propose A-ColViT, an adaptive user-interactive colorization framework to reduce the computational cost of vision transformers to be effectively used for real-time colorization applications. Given an input sample, A-ColViT is trained to satisfy the reconstruct error, and obtain desirable computational cost at the same time. An overview of our method is presented in Fig. 1.

Preliminaries. Vision transformer has demonstrated outstanding performance in computer vision tasks such as image classification. Thus, it has been actively adopted for colorization (Kumar, Weissenborn, and Kalchbrenner 2021; Huang, Zhao, and Liao 2022; Ji et al. 2022; Yun et al. 2022; Yin et al. 2021) task as well. Hence, we also adopt vision transformer architecture to propagate user hints

Given a colored train image $I_c \in \mathbb{R}^{H \times W \times 3}$, we convert the colored image to grayscale image, $I_g \in \mathbb{R}^{H \times W \times 1}$, by changing RGB color space to CIELab color space and extracting perceptual lightness value L^*. We generate user hints $I_{hint} \in \mathbb{R}^{H \times W \times 3}$ through masking the non-hint regions with 0 for a, b channels from L*a*b scale. The non-hint regions of a, b is combined with $I_{hint} \in \mathbb{R}^{H \times W \times 3}$ to compose I_{hint} where hint-regions have values of 1 and non-hint regions have values of 0. Since the users cannot directly interact in propagating hints during training, hints are uniformly sampled following 2-D gaussian distribution. The color variation of user hint is selected via taking the average color values for each channel in L*a*b color space adjacent to the hint region. Hence, the final output input $X \in \mathbb{R}^{H \times W \times 4}$ is obtained by concatenating grayscale image I_g and hint input I_{hint}. This final input is divided into

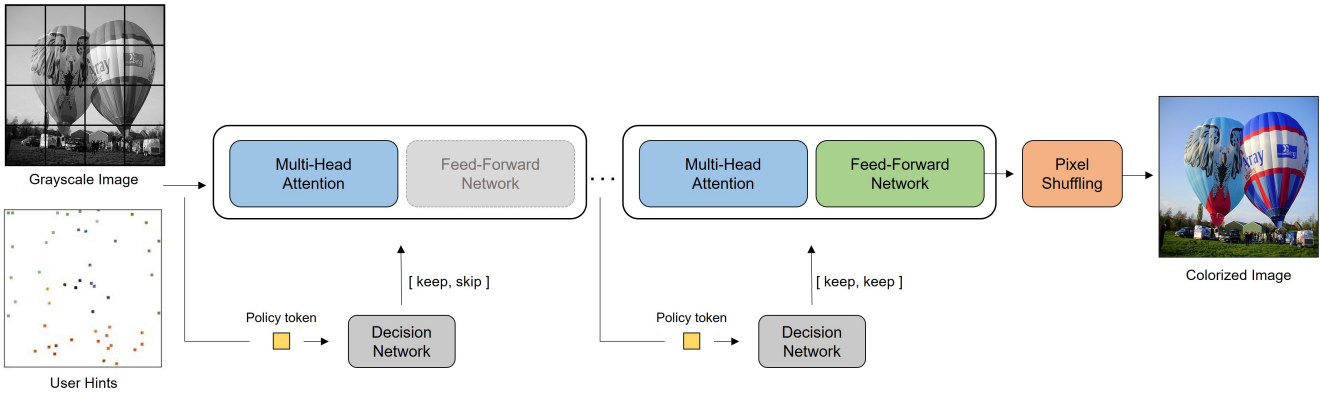

Figure 1: Overview of our proposed colorization pipeline. The main idea of our work is to use a decision network that uses a policy token in making a binary decision to dynamically skip or retain the attention layer in ViT. And, the output of transformer blocks are upsampled via pixel shuffling.

patches X_p that is fed into transformer encoder. The equation of obtained input X is defined as follows:

$$X = I_g \oplus I_{hint}, \tag{1}$$

where $\oplus$ is channel-wise concatenation.

Since the release of Transformers (Vaswani et al. 2017), numerous efforts have emphasized building different attention-based token mixer. However, in addition to having computational cost quadratic to the number of tokens to mix, self-attention and spatial MLPs bring significantly more parameters with long sequences, allowing them to only process hundreds of tokens. As a result, Poolformer (Yu et al. 2022) substitutes this attention-based token mixer with a straightforward operator, pooling, and takes advantage of pooling by adopting a hierarchical structure similar to conventional CNNs (He et al. 2016; Krizhevsky, Sutskever, and Hinton 2017; Simonyan and Zisserman 2014) and Transformers (Liu et al. 2021; Wang et al. 2021). Without any learnable parameters, the pooling requires a computational complexity linear to the sequence length, and the module aggregates each token with its nearby token features. As patch embedding of ViT, the model gets a sequence of embedded tokens $Z \in R^{N \times C}$ as input I, where N and C denote the sequence length and embedding dimension, respectively. The input of the model can be demonstrated as follows:

$$Z = [Z_{policy}; Z_1; Z_2; ...; Z_N] + E_{pos} \tag{2}$$

The single-head attention containing query, key, and value projected from the same input can be computed as below:

$$Attn(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

Multi-head self-attention concatenates the output from numerous single-head attentions and projects it with another parameter matrix to focus attention more efficiently on various representation subspaces:

$$head_{i,l} = Attn(Z_l W^Q_{i,l}, Z_l W^K_{i,l}, Z_l W^V_{i,l}) \tag{4}$$

$$MSA(Z_l) = Concat(head_{1,l}, ..., head_{H,l}) W^O_l, \tag{5}$$

where Z_l stands for the input at the l^{th} block and $W^Q_{i,l}, W^K_{i,l}, W^V_{i,l}$, and W^O_l are the parameter matrices in the i^{th} attention head of the l^{th} transformer block. The output of the MSA is fed into FFN, a two-layer MLP, to create the output of the transformer block Z_{l+1}. Residual connections are applied to MSA and FFN as follows:

$$Z'_l = MSA(Z_l) + Z_l, Z_{l+1} = FFN(Z'_l) + Z'_l \tag{6}$$

Using the class token from the previous transformer block (Z^0_L) as inputs, a linear layer generates the final prediction. By rearranging a (H/P, W/P, C×P^2) feature map into the shape of (H, W, C), we use pixel shuffling, an upsampling technique, to create a full-resolution image.

Decision Network. The decision network at l^{th} transformer block consists of two linear layers with parameters $W_l = \{W^a_l, W^f_l\}$ to produce usage policies for *attention layer selection* and *FFN layer selection*. Given the input to l^{th} block Z_l , the usage policy matrices for this block is computed as follows:

$$(m^a_l, m^f_l) = (W^a_l, W^f_l) Z_l \tag{7}$$

where m^a_l and m^f_l denote the usage policies of attention and FFN layers, respectively. Each m^a_l and m^f_l is passed forward to a *sigmoid* function, indicating the probability of keeping the corresponding attention and FFN layers, respectively. Thus, M^a_l and M^f_l to make decisions by sampling from m^a_l and m^f_l. In addition, since the binary decisions are non-differentiable, we adopt a Gumbel-Softmax trick (Maddison, Mnih, and Teh 2016) to enable backpropagation.

Layer Selection. When a transformer layer is redundant, that layer can be skipped. In this paper, we dynamically skip the attention layer and the FFN layer according to the input sample. The operation according to skip can be expressed as follows:

Table 1: Summary of our results on three benchmark datasets. For fair comparison, we compare tiny model of iColoriT with A-ColViT-T. And, we compare GFLOPs, PSNR@ 1, 5, and 10 and LPIPS@1, 5, and 10, respectively. The performance results worse than the baseline value are colored in blue, while those performing better than the baseline are colored in red.

Dataset	Method	GFLOPs	PSNR@1 ↑	LPIPS@1 ↓	PSNR@5 ↑	LPIPS@5 ↓	PSNR@10 ↑	LPIPS@10 ↓	PSNR@50 ↑	LPIPS@50 ↓
ImageNet-ctest10k	iColoriT-T	1.43	25.69	0.113	27.79	0.093	28.82	0.085	31.08	0.068
	A-ColViT-T-0.9	1.32 (7.7%↓)	25.83 (0.14↑)	0.111 (0.002↓)	27.93 (0.14↑)	0.091 (0.002↓)	28.96 (0.14↑)	0.083 (0.002↓)	31.23 (0.15↑)	0.067 (0.001↓)
	A-ColViT-T-0.7	1.07 (25.2%↓)	25.77 (0.08↑)	0.111 (0.002↓)	27.86 (0.07↑)	0.092 (0.000−)	28.90 (0.08↑)	0.083 (0.002↓)	31.16 (0.08↑)	0.067 (0.001↓)
	A-ColViT-T-0.5	0.79 (44.8%↓)	25.66 (0.03↓)	0.114 (0.001↑)	27.75 (0.04↓)	0.094 (0.002↑)	28.76 (0.06↓)	0.086 (0.001↑)	31.00 (0.08↓)	0.069 (0.001↑)
	A-ColViT-T-0.3	0.46 (67.8%↓)	25.34 (0.35↓)	0.120 (0.006↑)	27.42 (0.37↓)	0.098 (0.005↑)	28.43 (0.39↓)	0.090 (0.005↑)	30.67 (0.41↓)	0.072 (0.004↑)
Oxford 102flowers	iColoriT-T	1.43	20.31	0.213	23.31	0.151	24.67	0.130	27.30	0.095
	A-ColViT-T-0.9	1.32 (7.7%↓)	20.49 (0.18↑)	0.205 (0.008↓)	23.53 (0.22↑)	0.145 (0.006↓)	24.89 (0.22↑)	0.124 (0.006↓)	27.50 (0.20↑)	0.091 (0.004↓)
	A-ColViT-T-0.7	1.09 (23.8↓)	20.44 (0.13↑)	0.208 (0.005↓)	23.44 (0.13↑)	0.148 (0.003↓)	24.80 (0.13↑)	0.126 (0.004↓)	27.40 (0.10↑)	0.093 (0.002↓)
	A-ColViT-T-0.5	0.86 (39.9%↓)	20.31 (0.00−)	0.213 (0.000−)	23.28 (0.03↓)	0.151 (0.000−)	24.64 (0.03↓)	0.129 (0.001↑)	27.24 (0.06↓)	0.095 (0.00−)
	A-ColViT-T-0.3	0.49 (65.7%↓)	20.06 (0.25↓)	0.225 (0.012↑)	22.87 (0.44↓)	0.162 (0.002↑)	24.19 (0.48↓)	0.139 (0.009↑)	26.80 (0.50↓)	0.103 (0.008↑)
CUB-200	iColoriT-T	1.43	26.64	0.121	28.74	0.010	29.60	0.090	31.60	0.073
	A-ColViT-T-0.9	1.32 (7.7%↓)	26.77 (0.13↑)	0.118 (0.003↓)	28.86 (0.12↑)	0.096 (0.004↓)	29.71 (0.11↑)	0.088 (0.002↓)	31.74 (0.14↑)	0.071 (0.002↓)
	A-ColViT-T-0.7	1.07 (25.2%↓)	26.72 (0.08↑)	0.120 (0.001↓)	28.80 (0.06↑)	0.097 (0.003↓)	29.66 (0.06↑)	0.089 (0.001↓)	31.67 (0.07↑)	0.072 (0.001↓)
	A-ColViT-T-0.5	0.79 (44.8%↓)	26.62 (0.02↓)	0.121 (0.000−)	28.68 (0.06↓)	0.099 (0.001↓)	29.52 (0.08↓)	0.090 (0.000−)	31.50 (0.10↓)	0.074 (0.001↑)
	A-ColViT-T-0.3	0.46 (67.8%↓)	26.30 (0.34↓)	0.129 (0.008↑)	28.40 (0.34↓)	0.103 (0.003↑)	29.23 (0.37↓)	0.094 (0.004↑)	31.17 (0.43↓)	0.078 (0.005↑)

$$Z_l^{'} = M_{l,0}^{a} \cdot Attention(Z_l) + Z_l$$
$$Z_{l+1}^{'} = M_{l,1}^{f} \cdot FFN(Z_l^{'}) + Z_l^{'} \quad (8)$$

Loss function. Our goal is to optimize overall huber loss (Huber 1992) and the sparsity loss to train a vision transformer with an ideal target computational cost and minimal performance drop at the same time. The loss function of our A-ColViT can be defined as follows:

$$L_{sparsity} = (\frac{1}{L}\sum_{l=1}^{L} M_l^a - \beta_a)^2 + (\frac{1}{L}\sum_{l=1}^{L} M_l^f - \beta_f)^2 \quad (9)$$

$$\mathcal{L} = \mathcal{L}_{huber} + \mathcal{L}_{sparsity}, \quad (10)$$

where L, $\mathcal{L}_{huber}$, and $\mathcal{L}_{sparsity}$ represent the number of transformer layers, huber loss, and sparsity loss. Also, the hyperparameters β_a and β_f are target computation budgets with values between 0 and 1, which can adjust the remaining ratio of layers.

Experiments

Experimental settings. In training, we utilized a similar configuration with iColoriT (Yun et al. 2022) for equitable comparison. First, we resized our images to 224×224 with 512 batches and trained for 2.5M iterations. Second, we use patch size of $P = 16$ with sequence length N of 196. Moreover, we use AdamW optimizer (Loshchilov and Hutter 2017) with 0.0001 learning rate followed, a weight decay 0.05 and a cosine annealing scheduler (Loshchilov and Hutter 2016) for 25 epochs.

Baselines. We compare the performance of our model with iColoriT, a recent interactive colorization method based on Vision Transformer. Moreover, we compare our model with iColoriT-T, where T refers to tiny.

Datasets. To extensively explore model scalability, we utilize ILSVRC-2012 ImageNet dataset with 1.3M images and $1,000$ classes for training. We used 10,000 images for test set (also referred as ImageNet ctest10k). ImageNet ctest10k (Larsson, Maire, and Shakhnarovich 2016) is a subset of ImageNet that is used as a benchmark for colorization tasks. To further evaluate the performance of our model, we also selected CUB-200 dataset (Welinder et al. 2010) and Oxford 102 Flower dataset (Nilsback and Zisserman 2008) with $5,794$ test images of 200 classes and $1,000$ flower images of 102 classes, respectively.

Evaluation metric. To quantitatively evaluate the performance of our method, we measure and compare PSNR and learned perception image patch similarity LPIPS (Zhang et al. 2018) between the ground truth and the output image. PSNR of an image is defined as the ratio of an image's maximum achievable power to the power of corrupting noise. The higher the PSNR value means that the two images are alike (and share similar colors). LPIPS calculates perceptual similarity between two images like human perception. The lower the LPIPS value demonstrates that the two images are perceptual similar. Also, to evaluate and compare model efficiency, we mention then number of giga floating-point operations (GFLOPs).

Quantitative Results. In Table 1, we provide quantitative results in three datasets: ImageNet-ctest10k, Oxford 102flowers, and CUB-200, respectively. When we reduce GFLOPs to 0.3 (67.8% less than iColoriT-T), there is a slight performance degradation in PSNR@1 of 1.4% concerning iColoriT-T. In contrast, when we decrease our GFLOPs up to 0.5 or 50% reduction, there is almost no performance degradation compared to the baseline iColoriT-T. Moreover, for A-ColViT-T-0.9 and A-ColViT-T-0.7, the

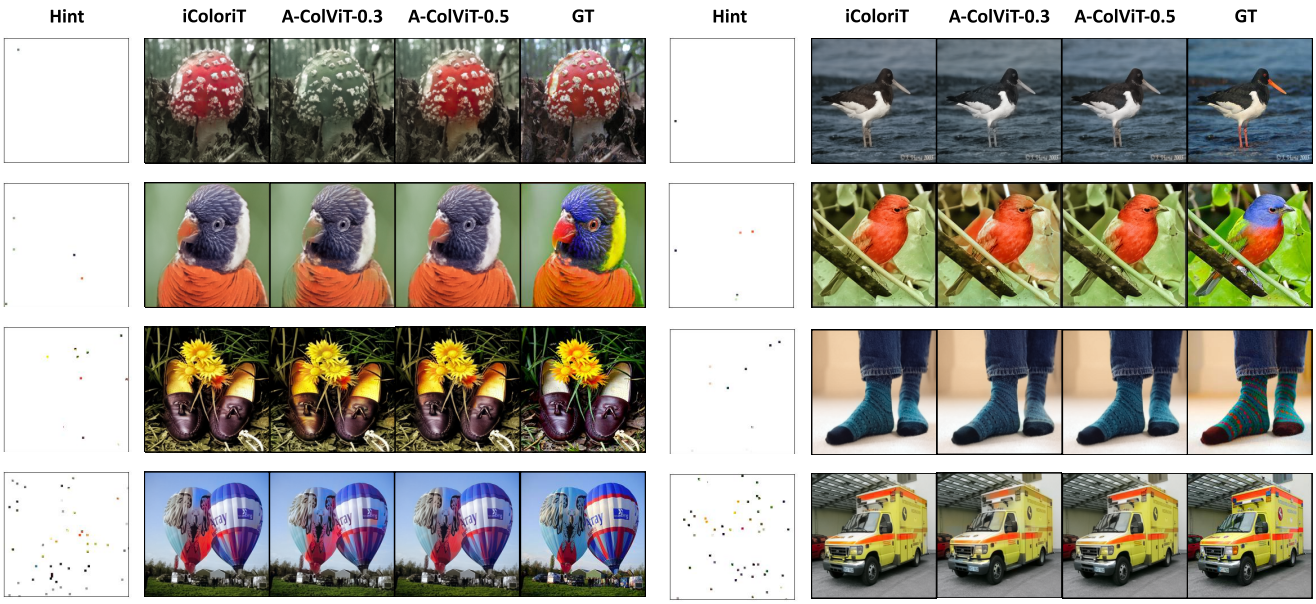

Figure 2: Qualitative visualization result of iColoriT, A-ColViT-0.3, A-ColViT-0.5 and ground truth image. In each row, the first square shows the number of hints used. We show the results on 1, 5, 10, and 100 number of user hints, respectively. As shown, A-ColViT-0.5 can generate images equivalent to iColoriT, despite the decreased GFLOPs.

PSNR@1 value increases by 0.5% and 0.3%, respectively. This demonstrates that despite the reduction of GFLOPs, there is a performance increase.

Qualitative Results. We provide the visualization of the qualitative results in Fig. 2. Given a test grayscale image, our goal is to reproduce a realistic colorized image that is equivalent to the ground truth. The results illustrate that the colorized output of ours does not differ with iColoriT with respect to the quality of the produced result. When we reduce GFLOPs significantly, A-ColViT-T-0.3 depicts less color in the output image. In comparison with A-ColViT-T-0.5, it shows analogous colors with the baseline. Moreover, our model is capable of colorizing detailed regions when adequate number of hints are provided as shown in the last row of Table 2. For some examples, A-ColViT-T-0.5 was able to colorize better. In the second row of Fig. 2, A-ColViT-T-0.5 was able to colorize detail regions (i.e. bird beak, bird body) in comparison with iColoriT.

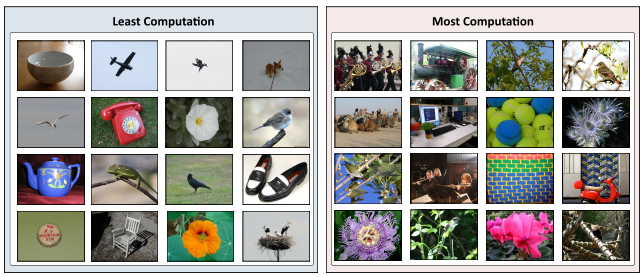

Figure 3: A set of sample images that require the least and most computation.

Table 2: Comparison between actual acceleration (Throughput and Speed up) and theoretical acceleration (GFLOPs) of iColoriT-T and A-ColViT-T.

Methods	Throughput (imgs/sec)	Speed up	GFLOPs
iColoriT-T	23.6	-	1.43
A-ColViT-T-0.3 (Ours)	50.8	2.2×	0.46
A-ColViT-T-0.5 (Ours)	40.0	1.7×	0.79

Discussion

Allocation of computational resources. To validate the result whether we have adjusted the computational cost appropriately for difficulty of each input, we visualize example images that take the least and most computation in Fig. 3. For least computed images, the images do not show diverse colors and illustrate only a single object (i.e. airplane, birds). Furthermore, background of least computed images are mostly white background or colors with less light intensity. For images that take most computation, multiple objects appear in a single image. For example, in the first two rows, images of British Royal Guard and people riding camels have multiple mixed objects with various colors. Flower images on the fourth row and the carpet tile on the third row have distinct colors with details. This makes the model more challenging resulting in taking more computation compared to the easy images.

Actual acceleration. Table. 2 demonstrates the Throughput and GFLOPs of iColoriT-T and A-ColViT. Throughput

indicates the number of processed images per second, measured by the CPU. To provide a fair comparison, we experimented with only single thread. A-ColViT-0.3 achieved 50.8 images per second with only 0.46 of GFLOPs, which is 1.7× faster than the comparison method iColoriT-T. In particular, A-ColViT-0.5 outperforms iColoriT-T by 2.2× faster, showing 40.0 throughput with only 0.79 GFLOPs, while requiring only a few performance drops. In conclusion, our method reduced theoretical FLOPs while achieving actual acceleration.

Conclusions

In this work, we present A-ColViT, an adaptive vision transformer for real-time interactive colorization. Our approach adaptively prunes vision transformer layers based on the difficulty of input samples. To achieve the efficiency and real-time colorization requirements, we use a trainable decision network to determine which specific layers to skip or retain in the transformer architecture. With the improved efficiency of the network obtained from the decision network, our experiments demonstrate that we are able to reduce the computational cost, while achieving and maintaining the SOTA performance.

References

Bai, Y.; Dong, C.; Chai, Z.; Wang, A.; Xu, Z.; and Yuan, C. 2022. Semantic-Sparse Colorization Network for Deep Exemplar-based Colorization. In *European Conference on Computer Vision*, 505–521. Springer.

Chen, C.-F. R.; Fan, Q.; and Panda, R. 2021. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, 357–366.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Gao, X.; Zhao, Y.; Dudziak, Ł.; Mullins, R.; and Xu, C.-z. 2018. Dynamic channel pruning: Feature boosting and suppression. *arXiv preprint arXiv:1810.05331*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

He, M.; Chen, D.; Liao, J.; Sander, P. V.; and Yuan, L. 2018. Deep exemplar-based colorization. *ACM Transactions on Graphics (TOG)*, 37(4): 1–16.

Ho, J.; Kalchbrenner, N.; Weissenborn, D.; and Salimans, T. 2019. Axial attention in multidimensional transformers. *arXiv preprint arXiv:1912.12180*.

Huang, Z.; Zhao, N.; and Liao, J. 2022. UniColor: A Unified Framework for Multi-Modal Colorization with Transformer. *arXiv preprint arXiv:2209.11223*.

Huber, P. J. 1992. Robust estimation of a location parameter. In *Breakthroughs in statistics*, 492–518. Springer.

Ji, X.; Jiang, B.; Luo, D.; Tao, G.; Chu, W.; Xie, Z.; Wang, C.; and Tai, Y. 2022. ColorFormer: Image Colorization via Color Memory Assisted Hybrid-Attention Transformer. In *European Conference on Computer Vision*, 20–36. Springer.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2017. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84–90.

Kumar, M.; Weissenborn, D.; and Kalchbrenner, N. 2021. Colorization transformer. *arXiv preprint arXiv:2102.04432*.

Larsson, G.; Maire, M.; and Shakhnarovich, G. 2016. Learning representations for automatic colorization. In *European conference on computer vision*, 577–593. Springer.

Levin, A.; Lischinski, D.; and Weiss, Y. 2004. Colorization using optimization. In *ACM SIGGRAPH 2004 Papers*, 689–694.

Li, Y.; Zhang, K.; Cao, J.; Timofte, R.; and Van Gool, L. 2021. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10012–10022.

Loshchilov, I.; and Hutter, F. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Maddison, C. J.; Mnih, A.; and Teh, Y. W. 2016. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.

Meng, L.; Li, H.; Chen, B.-C.; Lan, S.; Wu, Z.; Jiang, Y.-G.; and Lim, S.-N. 2022. AdaViT: Adaptive Vision Transformers for Efficient Image Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12309–12318.

Nilsback, M.-E.; and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, 722–729. IEEE.

Rao, Y.; Zhao, W.; Liu, B.; Lu, J.; Zhou, J.; and Hsieh, C.-J. 2021. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34: 13937–13949.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Veit, A.; and Belongie, S. 2018. Convolutional networks with adaptive inference graphs. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 3–18.

Wang, W.; Xie, E.; Li, X.; Fan, D.-P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; and Shao, L. 2021. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 568–578.

Wang, X.; Yu, F.; Dou, Z.-Y.; Darrell, T.; and Gonzalez, J. E. 2018. Skipnet: Learning dynamic routing in convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 409–424.

Welinder, P.; Branson, S.; Mita, T.; Wah, C.; Schroff, F.; Belongie, S.; and Perona, P. 2010. Caltech-UCSD birds 200.

Wu, Z.; Nagarajan, T.; Kumar, A.; Rennie, S.; Davis, L. S.; Grauman, K.; and Feris, R. 2018. Blockdrop: Dynamic inference paths in residual networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8817–8826.

Yin, H.; Gong, Y.; and Qiu, G. 2019. Side window filtering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8758–8766.

Yin, H.; Vahdat, A.; Alvarez, J. M.; Mallya, A.; Kautz, J.; and Molchanov, P. 2022. A-ViT: Adaptive Tokens for Efficient Vision Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10809–10818.

Yin, W.; Lu, P.; Zhao, Z.; and Peng, X. 2021. Yes," Attention Is All You Need", for Exemplar based Colorization. In *Proceedings of the 29th ACM International Conference on Multimedia*, 2243–2251.

Yu, W.; Luo, M.; Zhou, P.; Si, C.; Zhou, Y.; Wang, X.; Feng, J.; and Yan, S. 2022. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10819–10829.

Yun, J.; Lee, S.; Park, M.; and Choo, J. 2022. iColoriT: Towards Propagating Local Hint to the Right Region in Interactive Colorization by Leveraging Vision Transformer. *arXiv preprint arXiv:2207.06831*.

Zhang, L.; Ji, Y.; Lin, X.; and Liu, C. 2017a. Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan. In *2017 4th IAPR Asian conference on pattern recognition (ACPR)*, 506–511. IEEE.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 586–595.

Zhang, R.; Zhu, J.-Y.; Isola, P.; Geng, X.; Lin, A. S.; Yu, T.; and Efros, A. A. 2017b. Real-time user-guided image colorization with learned deep priors. *arXiv preprint arXiv:1705.02999*.