Colorization Transformer

(ICLR 2021)

Transformers for diverse and high-resolution image colorization

Manoj Kumar

mechcoder@google.com

Dirk Weissenborn

diwe@google.com

Nal Kalchbrenner

nalk@google.com

Google Research

Colorization

Gray image

Google Research

Colorization

Gray image

ColTran Colorizations

Research

Colorization Transformer

- First application of transformers for 256x256 image colorization
- **Self Attention -** Global interactions between pixels
- Generating 256x256x3 symbols token-by-token is painfully slow and expensive!
- Decompose into 3 subtasks, trained in parallel

Google Research

High Level Approach

Model 1: Colorizer

- 64x64 coarse colorization
- Autoregressive sample

Inputs

Output

Google Research

High Level Approach

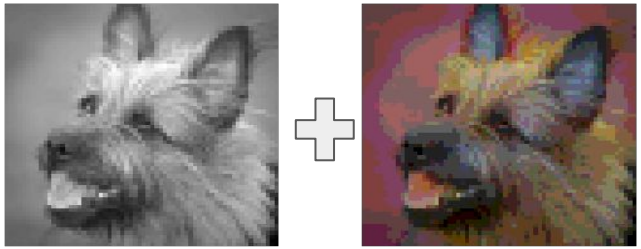
Model 1: Colorizer

- 64x64 coarse colorization
- Autoregressive sample

Model 2: Color upsampler

- Refine: 64x64 RGB image
- Deterministic model

Inputs



Output

Google Research

High Level Approach

Model 1: Colorizer

- 64x64 coarse colorization
- Autoregressive sample

Model 2: Color upsampler

- Refine: 64x64 RGB image
- Deterministic model

Model 3: Spatial upsampler

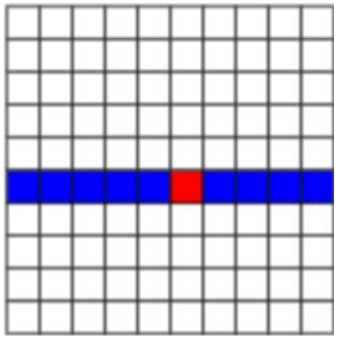
- Super resolve: 256x256 RGB image
- Deterministic model

Inputs

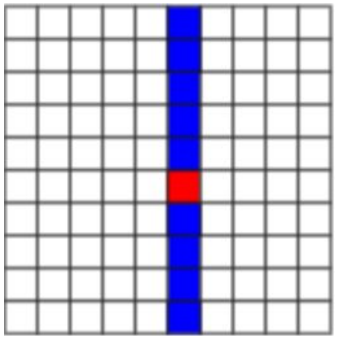
Output

Building Blocks: Axial Attention

Row attention



Column Attention

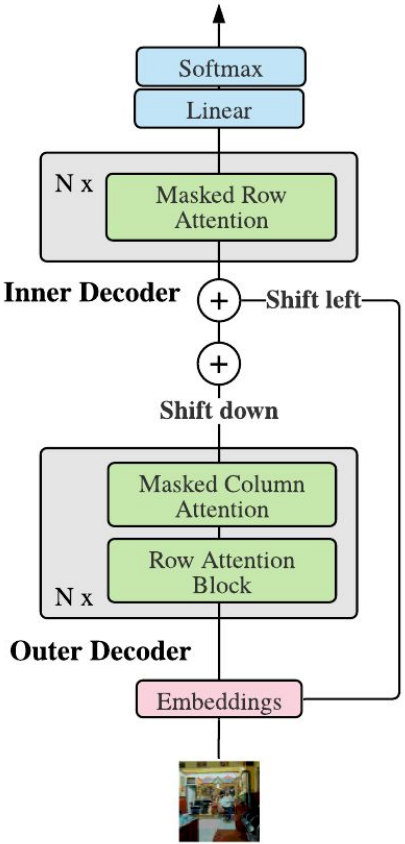


(Ho et al 2019, Wang et al 2020)

- Self-attention to each axis independently.
- Complexity reduction by a factor of **H**

Google Research

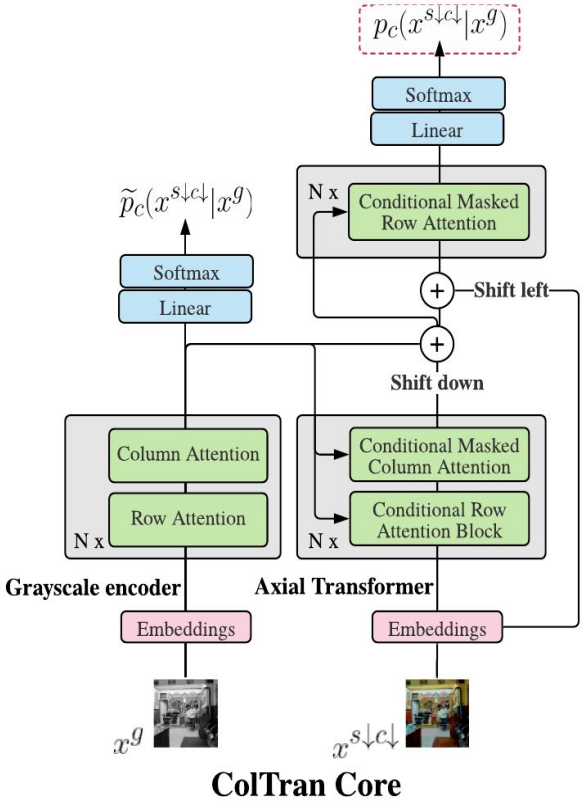
Building Blocks: Axial Transformer



- Autoregressive image generation model.
- Attends to all previous pixels as per raster order
- Natively supports semi-parallel sampling.

Google Research

ColTran Core: Autoregressive Colorizer



ColTran Core

Target discretization

- 3-bit RGB, with 8 colors per {R,G,B}.
- Total of 512 colors.

Grayscale encoder:

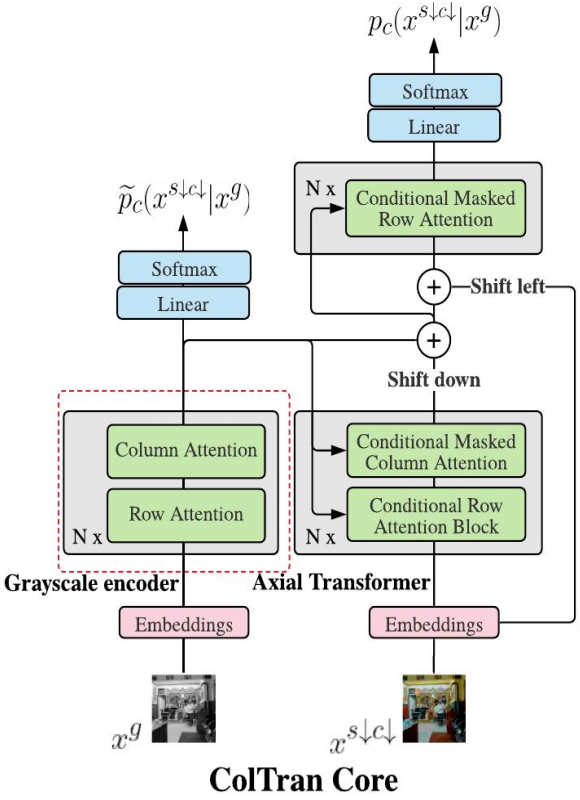
- Stack of axial attention layers.
- Captures context from the grayscale image

Auxiliary Parallel Model:

- Applied at the output of the encoder
- Models each color independently

Google Research

ColTran Core: Autoregressive Colorizer



ColTran Core

Target discretization

- 3-bit RGB, with 8 colors per {R,G,B}.
- Total of 512 colors.

Grayscale encoder:

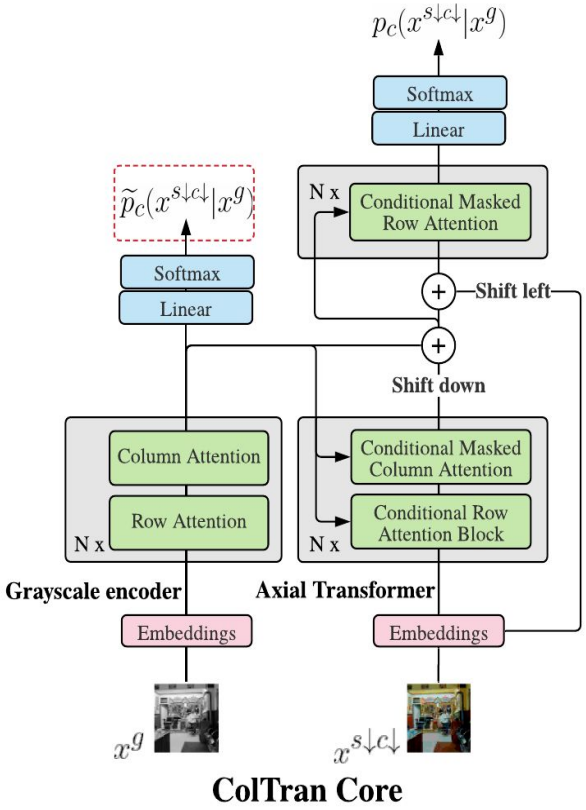
- Stack of axial attention layers.
- Captures context from the grayscale image

Auxiliary Parallel Model:

- Applied at the output of the encoder
- Models each color independently

Google Research

ColTran Core: Autoregressive Colorizer



Target discretization

- 3-bit RGB, with 8 colors per {R,G,B}.
- Total of 512 colors.

Grayscale encoder:

- Stack of axial attention layers.
- Captures context from the grayscale image

Auxiliary Parallel Model:

- Applied at the output of the encoder
- Models each color independently

Google Research

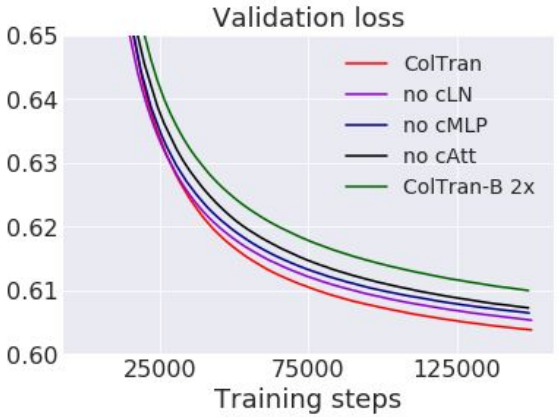
Conditional Transformer Layers

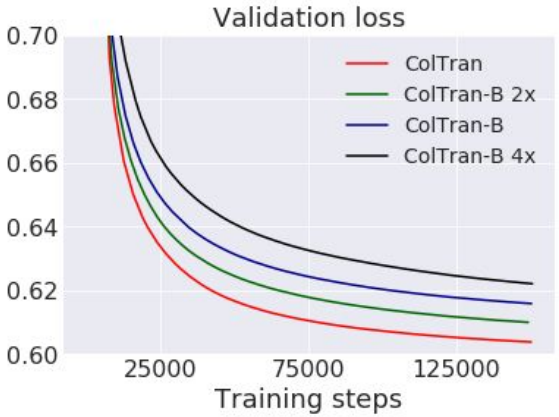
Component	Unconditional	Conditional
Self-Attention	$\mathbf{y} = \text{Softmax}(\frac{\mathbf{q}\mathbf{k}^\top}{\sqrt{D}})\mathbf{v}$	$\mathbf{y} = \text{Softmax}(\frac{\mathbf{q}_c\mathbf{k}_c^\top}{\sqrt{D}})\mathbf{v}_c$ where $\forall \mathbf{z} = \mathbf{k}, \mathbf{q}, \mathbf{v}$ $\mathbf{z}_c = (\mathbf{c}U_s^z) \odot \mathbf{z} + (\mathbf{c}U_b^z)$
MLP	$\mathbf{y} = \text{ReLU}(\mathbf{x}U_1 + \mathbf{b}_1)U_2 + \mathbf{b}_2$	$\mathbf{h} = \text{ReLU}(\mathbf{x}U_1 + \mathbf{b}_1)U_2 + \mathbf{b}_2$ $\mathbf{y} = (\mathbf{c}U_s^f) \odot \mathbf{h} + (\mathbf{c}U_b^f)$
Layer Norm	$\mathbf{y} = \beta\,\text{Norm}(\mathbf{x}) + \gamma$	$\mathbf{y} = \beta_c\,\text{Norm}(\mathbf{x}) + \gamma_c$ where $\forall \mu = \beta_c, \gamma_c$ $\mathbf{c} \in \mathbb{R}^{H\times W\times D} \rightarrow \hat{\mathbf{c}} \in \mathbb{R}^{HW\times D}$ $\mu = (\mathbf{u}\cdot\hat{\mathbf{c}})U_d^\mu \quad \mathbf{u} \in \mathbb{R}^{HW}$

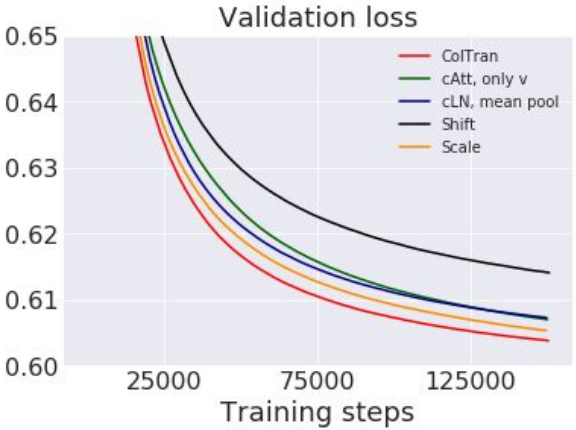
Table 1: We contrast the different components of unconditional self-attention with self-attention conditioned on context $\mathbf{c} \in \mathbb{R}^{M\times N\times D}$. Learnable parameters specific to conditioning are denoted by $\mathbf{u}$ and $U_\cdot \in \mathbb{R}^{D\times D}$.

Google Research

Conditional Transformer Layers

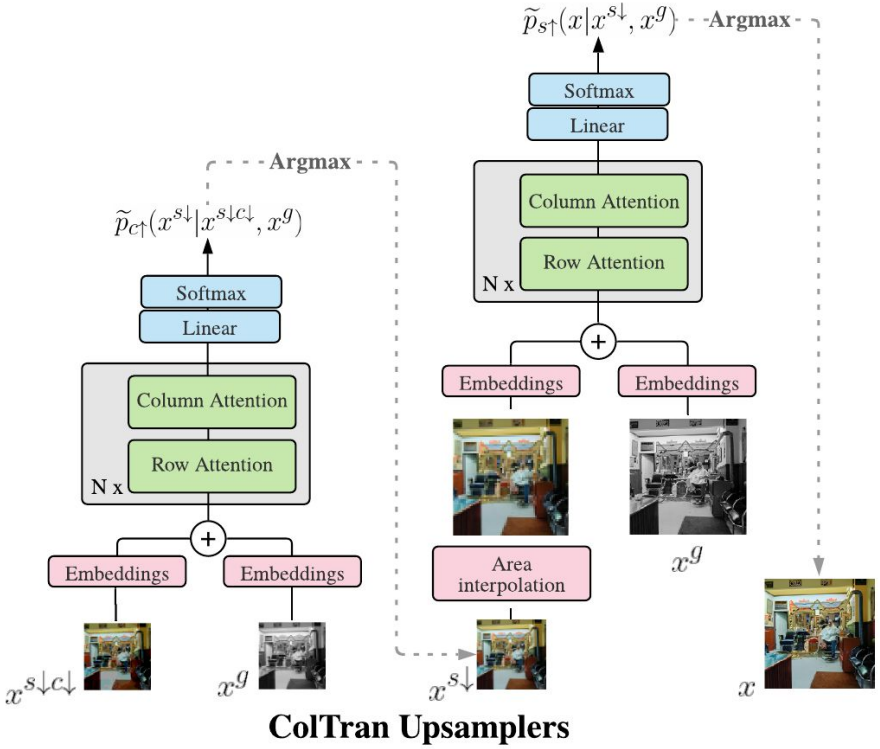






Google Research

ColTran Upsamplers

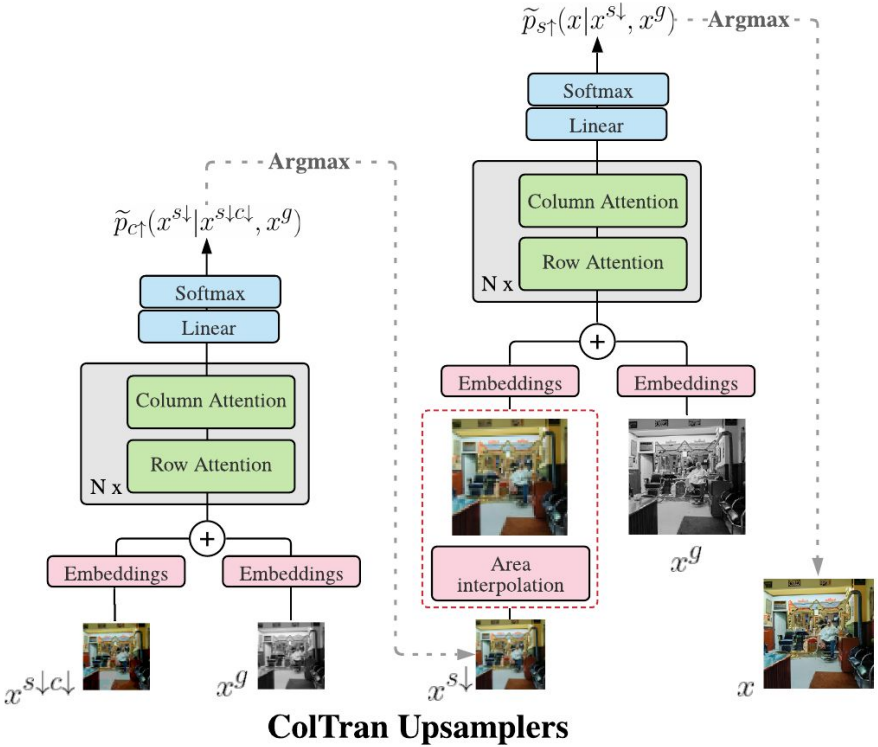


ColTran Upsamplers

- Upsamplers share similar architecture.
  - Stack of axial attention layers
- **Output:**
  - Per-pixel distribution over 256 intensities
- **Spatial Upsampler**: Single additional upsampling layer at the input.

Google Research

ColTran Upsamplers



ColTran Upsamplers

- Upsamplers share similar architecture.
  - Stack of axial attention layers
- **Output:**
  - Per-pixel distribution over 256 intensities
- **Spatial Upsampler**: Single additional upsampling layer at the input.

Google Research

Results

Models	FID
ColTran	**19.37 ± 0.09**
ColTran-B	19.98 ± 0.20
ColTran-S	22.06 ± 0.13
PixColor [16]	24.32 ± 0.21
cGAN [3]	24.41 ± 0.27
cINN [1]	25.13 ± 0.3
VAE-MDN [11]	25.98 ± 0.28
Ground truth	14.68 ± 0.15
Grayscale	30.19 ± 0.1

Models	AMT Fooling rate
ColTran (Oracle)	62.0 % ± 0.99
ColTran (Seed 1)	40.5 % ± 0.81
ColTran (Seed 2)	**42.3 % ± 0.76**
ColTran (Seed 3)	41.7 % ± 0.83
PixColor [16] (Oracle)	38.3 % ± 0.98
PixColor (Seed 1)	33.3 % ± 1.04
PixColor (Seed 2)	35.4 % ± 1.01
PixColor (Seed 3)	33.2 % ± 1.03
CIC [56]	29.2 % ± 0.98
LRAC [27]	30.9 % ± 1.02
LTBC [22]	25.8 % ± 0.97

Table 2: We outperform various state-of-the-art colorization models both on FID (left) and human evaluation (right). We obtain the FID scores from (Ardizzone et al., 2019) and the human evaluation results from (Guadarrama et al., 2017). ColTran-B is a baseline Axial Transformer that conditions via addition and ColTran-S is a control experiment where we train ColTran core (See: 4.1) on smaller 28×28 colored images.

Open source code

https://github.com/google-research/google-research/tree/master/coltran

Questions

mechcoder@google.com

Paper with more samples and insights

https://arxiv.org/abs/2102.04432

Google Research