

# Conditional Image Generation with PixelCNN Decoders

Aaron van den Oord<sup>1</sup>   Nal Kalchbrenner<sup>1</sup>   Oriol Vinyals<sup>1</sup>   Lasse  
Espeholt<sup>1</sup>   Alex Graves<sup>1</sup>   Koray Kavukcuoglu<sup>1</sup>

<sup>1</sup>Google DeepMind

NIPS, 2016

Presenter: Beilun Wang

## 1 Introduction

- Motivation
- Previous Solutions
- Contributions

## 2 Proposed Methods

- PixelCNN
- Conditional PixelCNN

## 3 Summary

## 1 Introduction

- Motivation
- Previous Solutions
- Contributions

## 2 Proposed Methods

- PixelCNN
- Conditional PixelCNN

## 3 Summary

## Motivation:

- Conditional image generator with image density model.
- This generator can also be conditioned on class labels, descriptions and a single human face.
- Fast and parallel training.

# Problem Setting:

## Problem Setting:

- Input: Image with missing pixels, OR image class labels, OR a vector in the embedded space, OR a single human face
- Target: joint distribution consisting of conditional distribution with CNN.
- Output: Image
- PixelCNN (Pixel RNN):

$$p(\mathbf{x}) = \prod_{i=1}^{n^2} p(x_i | x_1, \dots, x_{i-1}) \quad (1)$$

- Conditional Version:

$$p(\mathbf{x}|\mathbf{h}) = \prod_{i=1}^{n^2} p(x_i | x_1, \dots, x_{i-1}, \mathbf{h}) \quad (2)$$

- B conditioned on (R,G); G conditioned on R.

## 1 Introduction

- Motivation
- Previous Solutions
- Contributions

## 2 Proposed Methods

- PixelCNN
- Conditional PixelCNN

## 3 Summary

# Previous Solutions

- PixelRNN

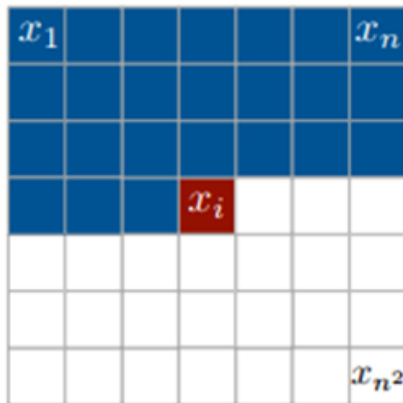


Figure: PixelRNN

## 1 Introduction

- Motivation
- Previous Solutions
- **Contributions**

## 2 Proposed Methods

- PixelCNN
- Conditional PixelCNN

## 3 Summary



# Contributions

- A fast and parallel trainable deep neural nets model for conditional image generator (?)
- Gated convolutional layers
- Conditional Gated convolutional layers

- 1 Introduction
  - Motivation
  - Previous Solutions
  - Contributions
- 2 Proposed Methods
  - PixelCNN
  - Conditional PixelCNN
- 3 Summary

# Pixel CNN

- Input:  $N \times N \times 3$
- Output:  $N \times N \times 3 \times 256$

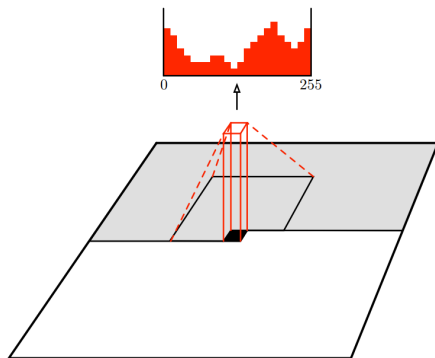


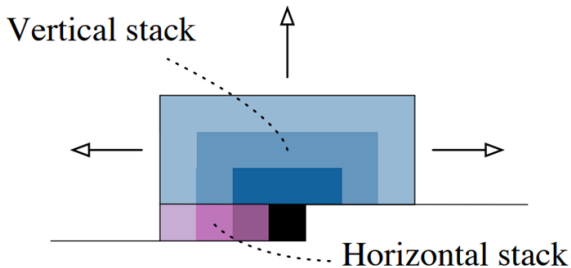
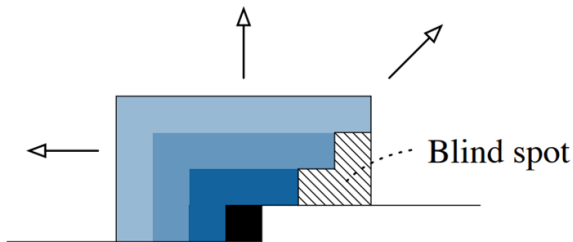
Figure: PixelCNN

Only above and left Pixels are considered.

# Masked Filter

1	1	1	1	1
1	1	1	1	1
1	1	0	0	0
0	0	0	0	0
0	0	0	0	0

# Blind spot



# Gated Convolutional Layers

- The gates in LSTM may help it to model more complex interactions.
- This is also studied by paper like Highway networks, grid LSTM, and Neural GPUs.

$$\mathbf{y} = \tanh(W_{k,f} * \mathbf{x}) \odot \sigma(W_{k,g} * \mathbf{x}) \quad (3)$$

# Gated Convolutional Layers – figure

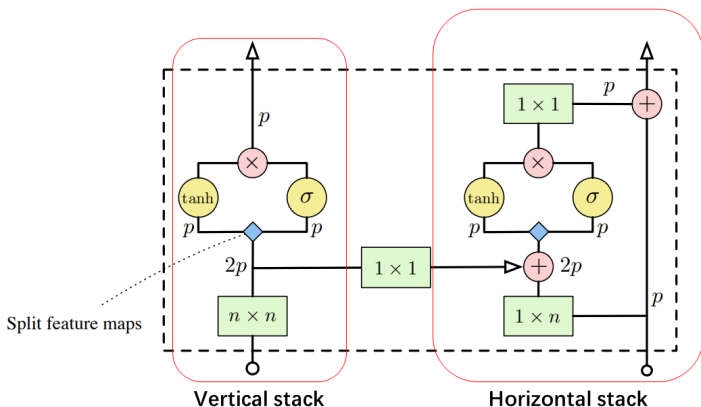


Figure: Gated Convolutional Layers

- 1 Introduction
  - Motivation
  - Previous Solutions
  - Contributions
- 2 Proposed Methods
  - PixelCNN
  - Conditional PixelCNN
- 3 Summary



- replace  $p(\mathbf{x}|\mathbf{h})$  from  $p(\mathbf{x})$ .

$$\mathbf{y} = \tanh(W_{k,f} * \mathbf{x} + V_{k,f}^T \mathbf{h}) \odot \sigma(W_{k,g} * \mathbf{x} + V_{k,g}^T \mathbf{h}) \quad (4)$$

- Use a deconvolutional neural nets  $m()$
- map  $\mathbf{h}$  back to the image space as  $\mathbf{s}$

$$\mathbf{y} = \tanh(W_{k,f} * \mathbf{x} + V_{k,f}^T \mathbf{s}) \odot \sigma(W_{k,g} * \mathbf{x} + V_{k,g}^T \mathbf{s}) \quad (5)$$

# Experiment Results—image generation based on labels



African elephant

Coral Reef



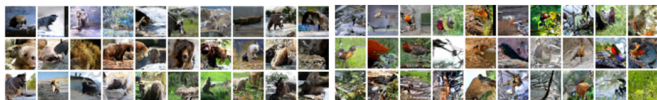
Sandbar

Sorrel horse



Lhasa Apso (dog)

Lawn mower



Brown bear

Robin (bird)

# Experiment Results—image generation based on a single human face



# Summary

- This paper improves the PixelCNN by the gated activation unit
- This paper extends the PixelCNN to a conditional version