Pixel Recurrent Neural Network

Пальчиков Николай

162

Generative image modelling

- Inpainting
- Deblurring
- Image compressions
- Generation of new images

occluded completions original

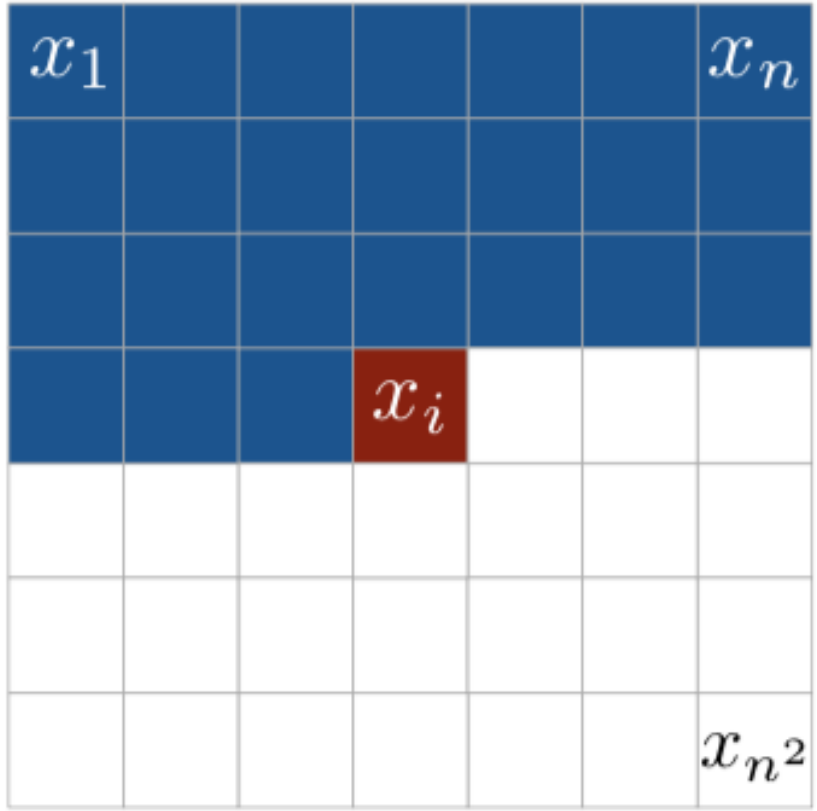
Формализуем задачу

- Пусть x – картинка, $x \in \mathbb{R}^{n \times n}$ Хотим смоделировать плотность.

$$p(x) = \prod_{i=1}^{n \times n} p(x_i \mid x_{i-1} \dots x_1)$$

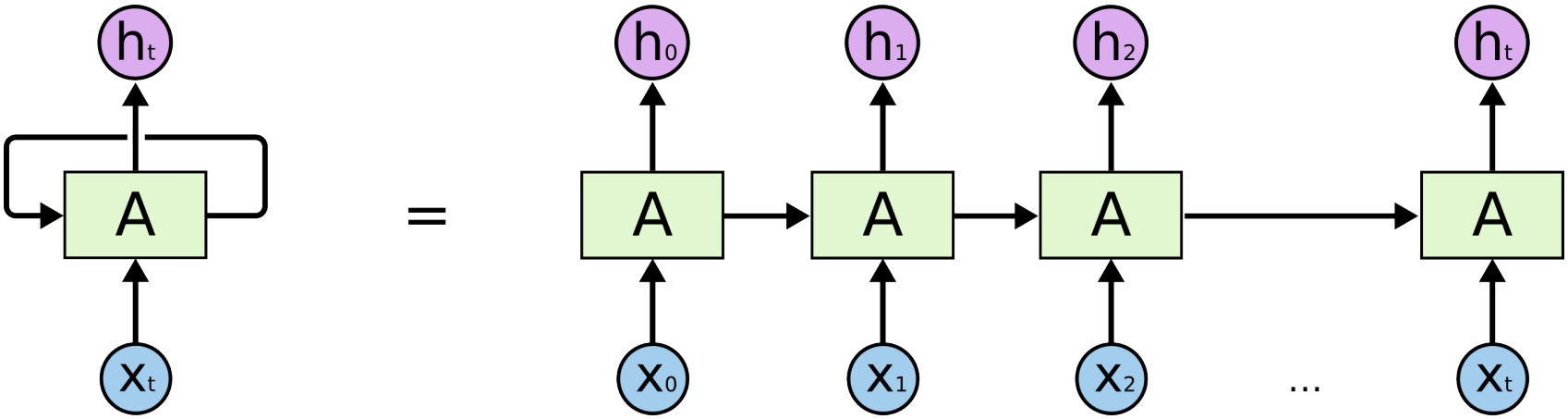
- Учитывая тот факт, что картинка может быть цветной, каждый множитель $p(x_i \mid x_{<i})$ можно записать в виде

$$p(x_i \mid x_{<i}) = p(x_{i,R} \mid x_{<i}) \cdot p(x_{i,G} \mid x_{i,R}, x_{<i}) \cdot p(x_{i,B} \mid x_{i,G}, x_{i,R}, x_{<i})$$

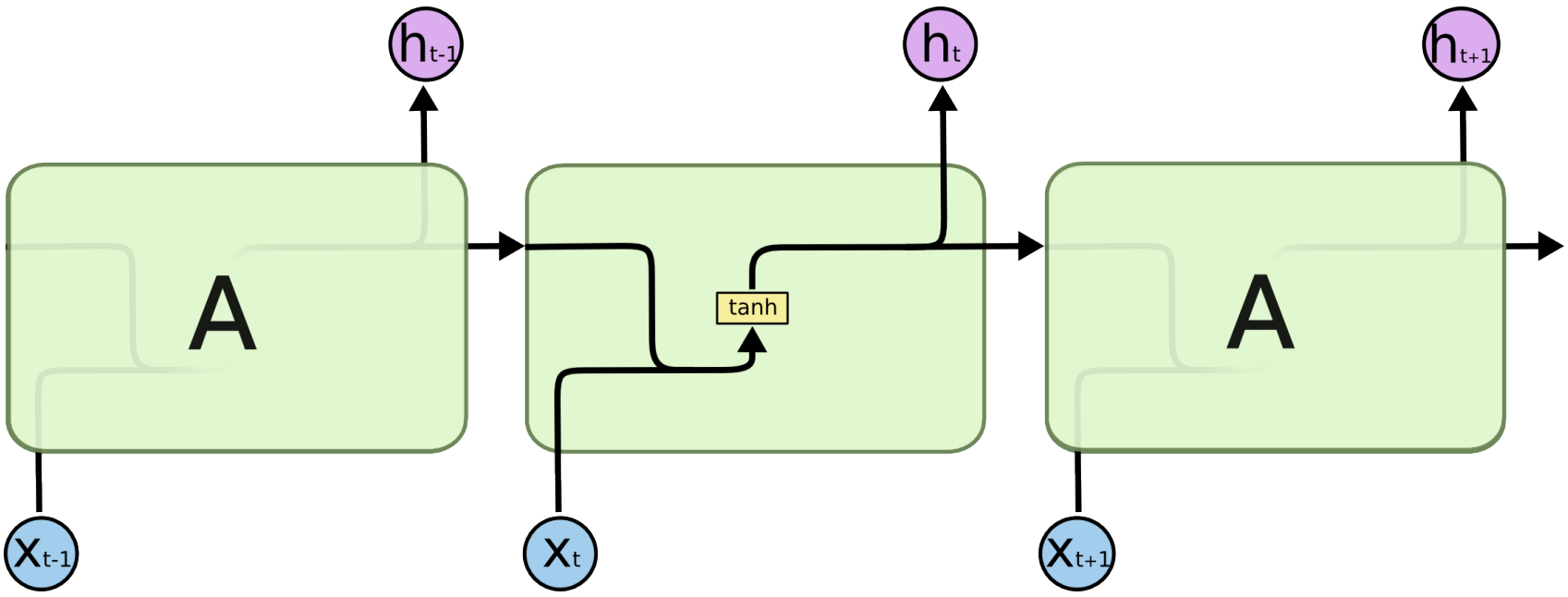


- Получаем задачу вида "сгенерировать следующий элемент последовательности, зная предыдущие"
- Считаем, что пиксели принимают целочисленные значения от 0 до 255
- Рекуррентные нейронные сети неплохо справляются с подобными.

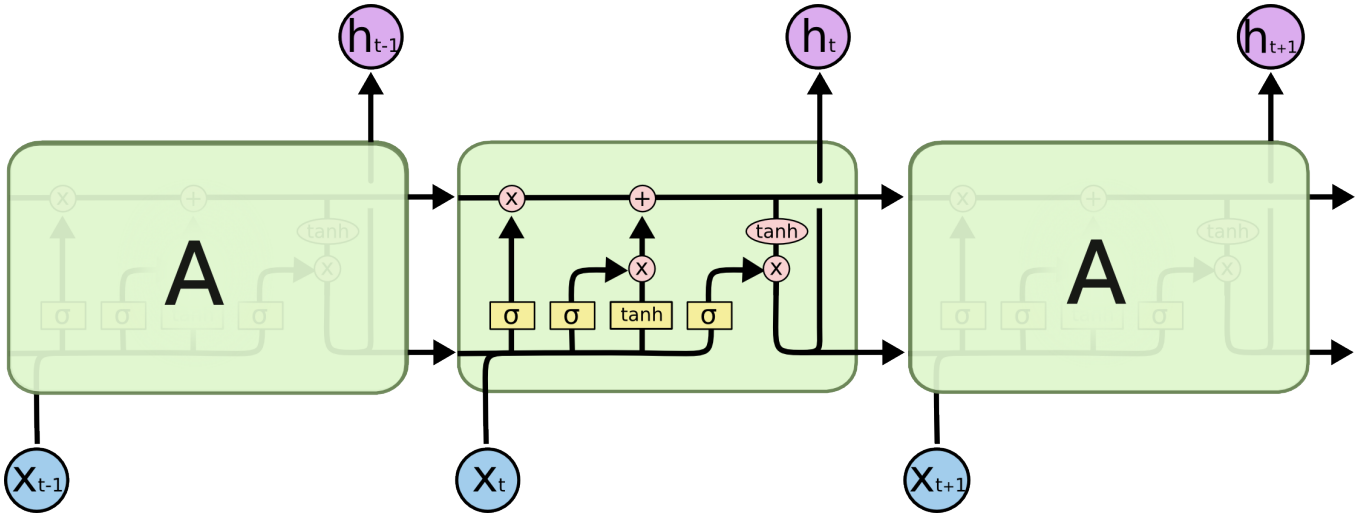
RNN



RNN

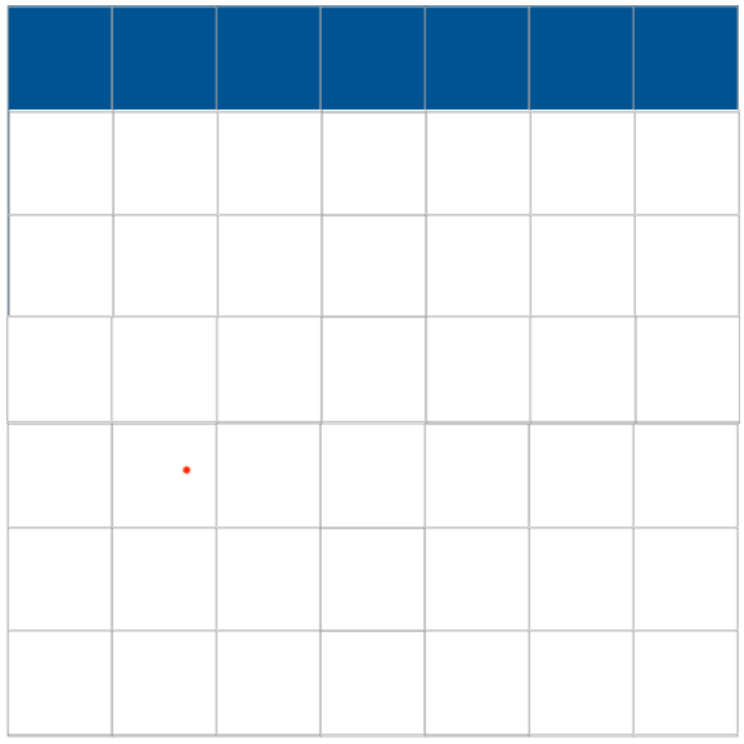


LSTM

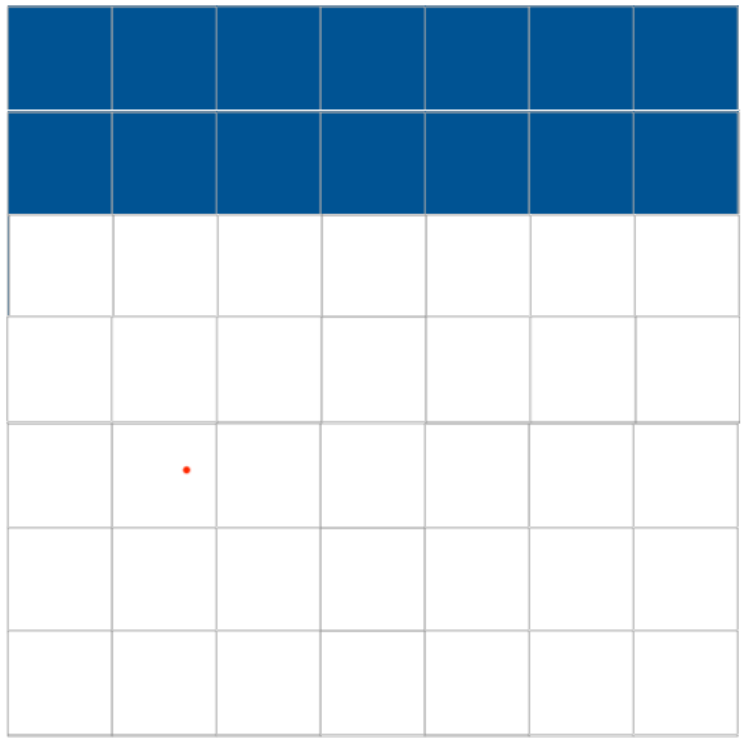


$$\mathrm{i} = \sigma(x_i \cdot U^i + h_{i-1} \cdot W^i)$$
$$\mathrm{f} = \sigma(x_i \cdot U^f + h_{i-1} \cdot W^f)$$
$$\mathrm{o} = \sigma(x_i \cdot U^o + h_{i-1} \cdot W^o)$$
$$\mathrm{g} = \tanh(x_i \cdot U^g + h_{i-1} \cdot W^g)$$
$$\mathrm{c}_i = c_{i-1} \odot f + g \odot i$$
$$\mathrm{h}_i = \tanh(c_i) \cdot o_i$$

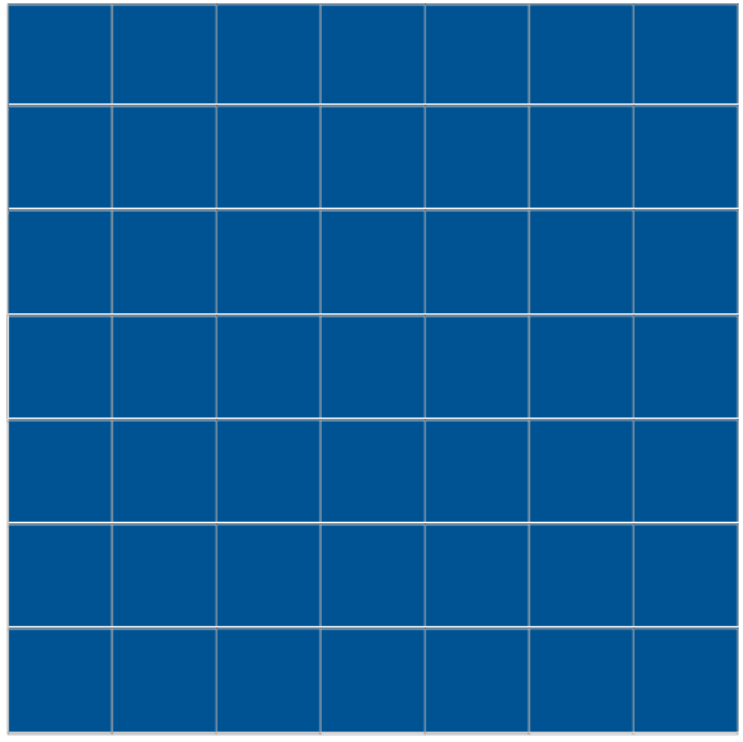
Row LSTM



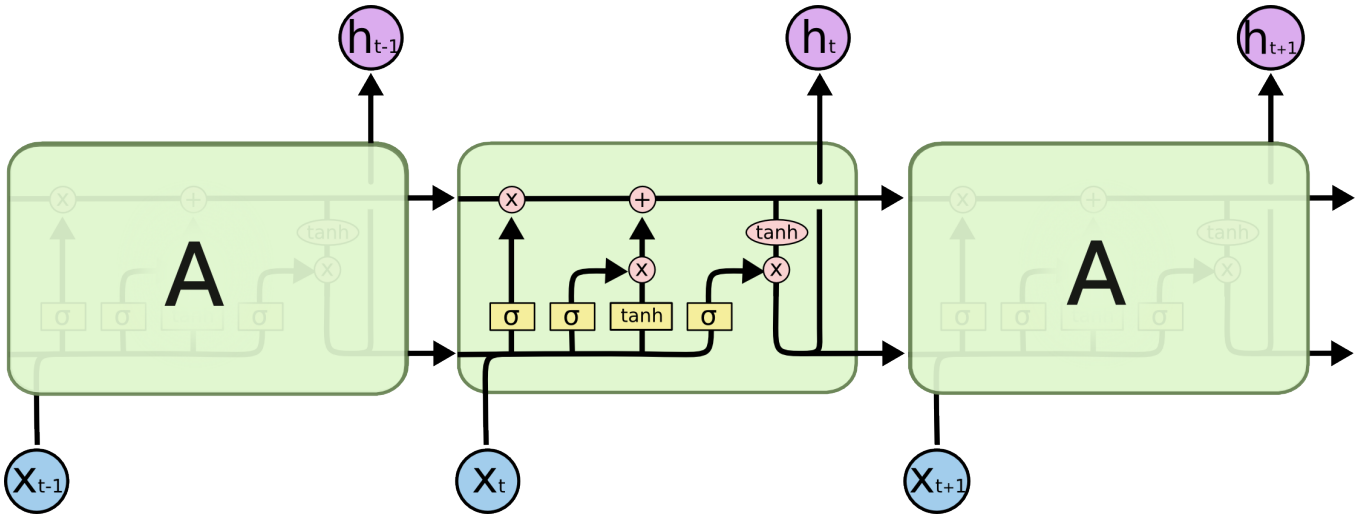
Row LSTM



Row LSTM



LSTM + Свертка

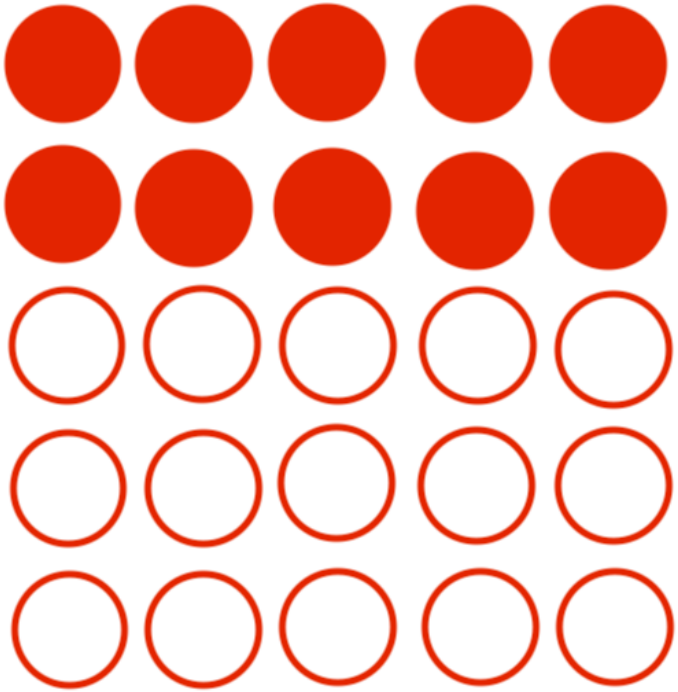


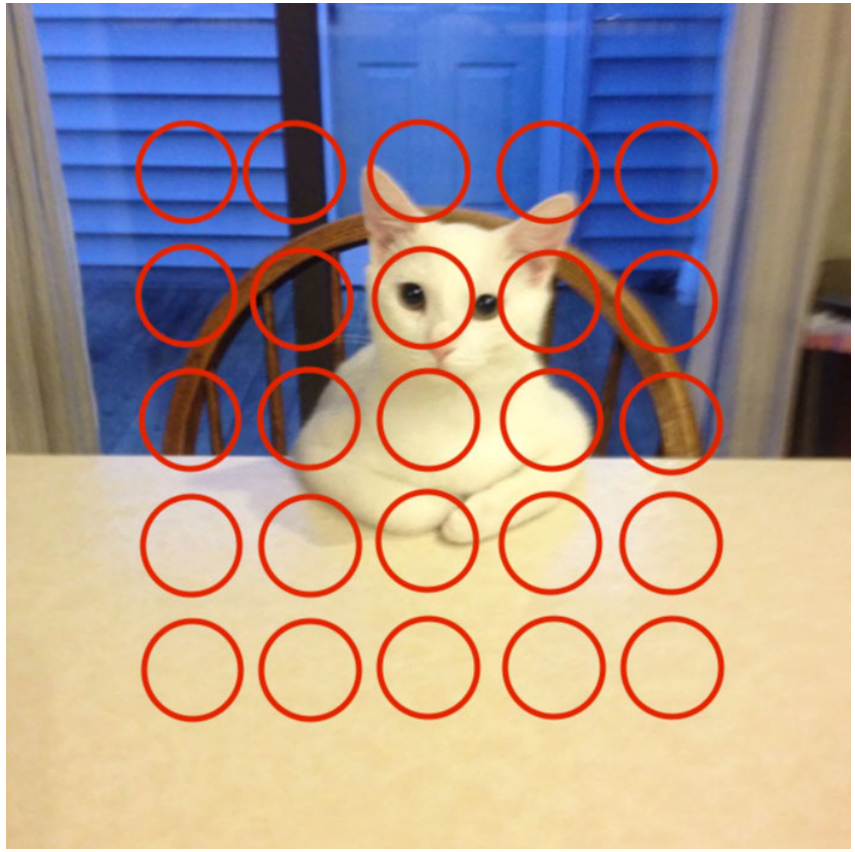
$$o_i, f_i, i_i, g_i = \sigma(K^{ss} \circledast h_{i-1} + K^{is} \circledast x_i)$$
$$c_i = f_i \odot c_{i-1} + i_i \odot g_i$$
$$h_i = \tanh(c_i) \odot o_i$$

Input-to-state

Нужно посчитать:

$$\sigma(K^{ss} \circledast h_{i-1} + K^{is} \circledast x_i)$$

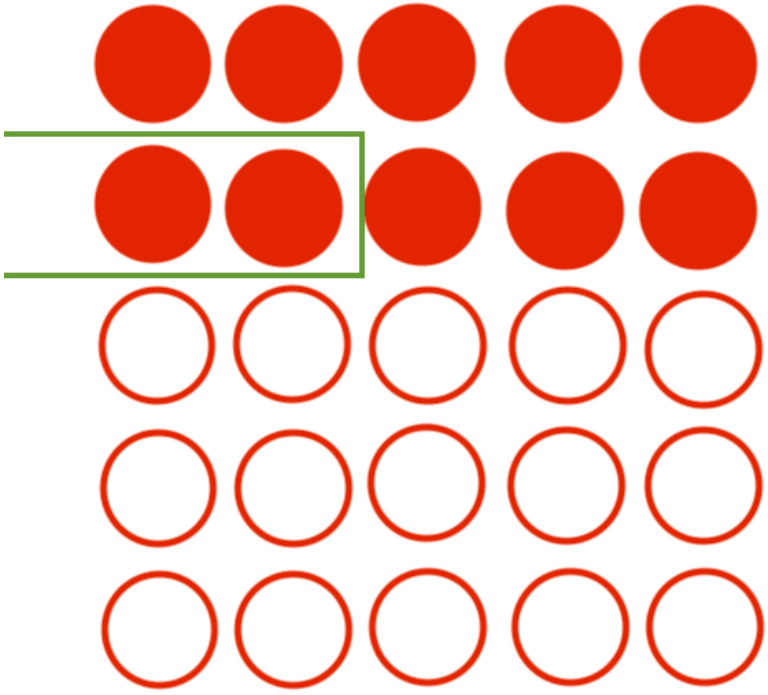


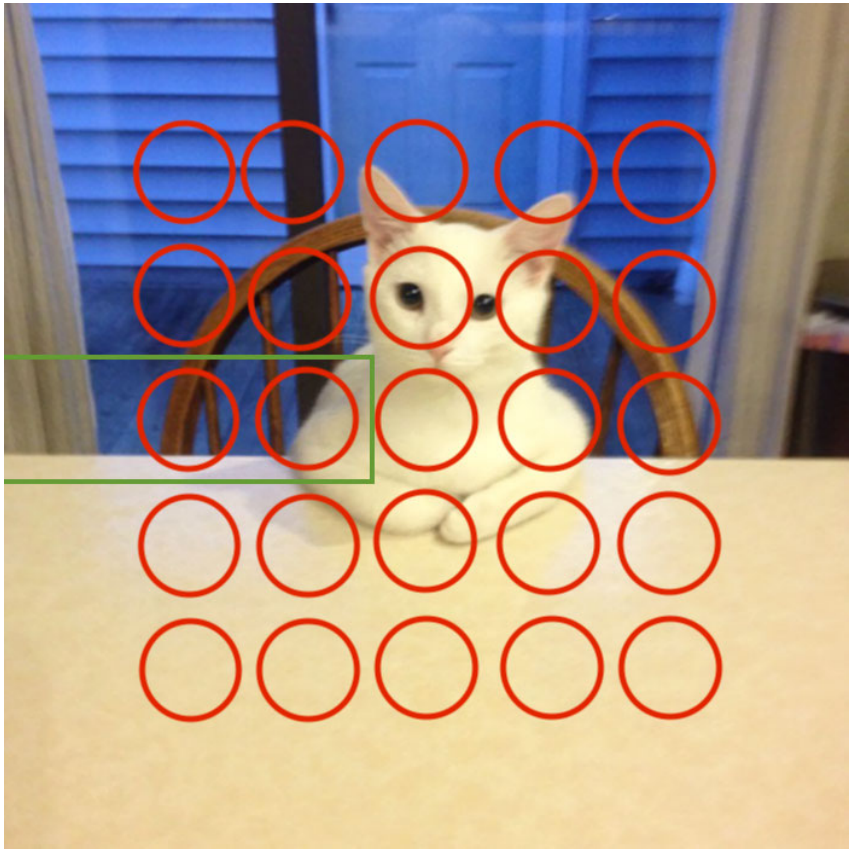


Input-to-state

Нужно посчитать:

$$\sigma(K^{ss} \circledast h_{i-1} + K^{is} \circledast x_i)$$

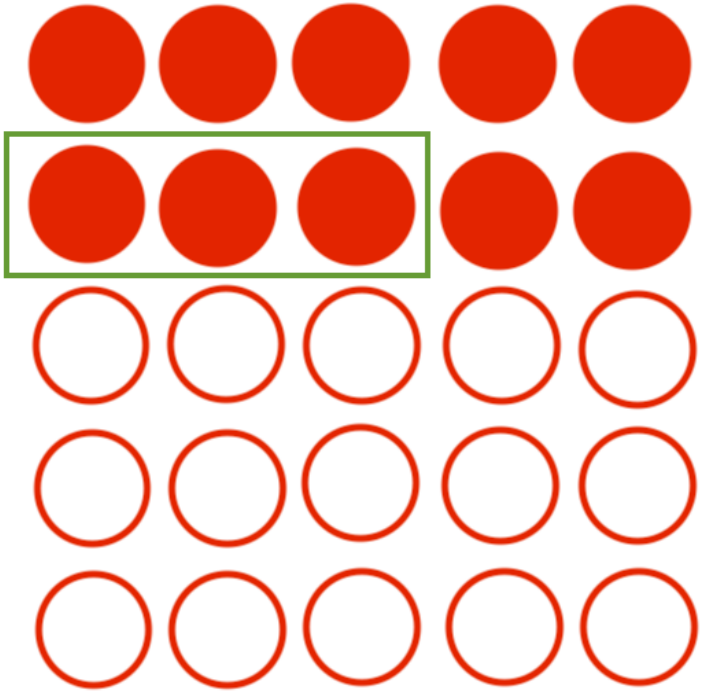


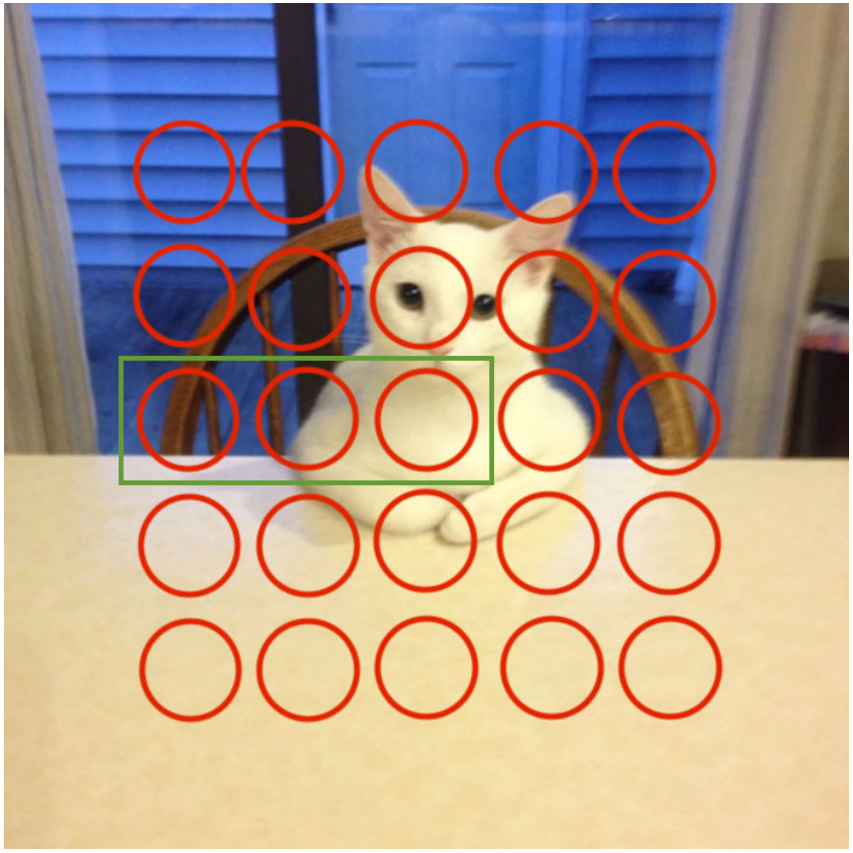


Input-to-state

Нужно посчитать:

$$\sigma(K^{ss} \circledast h_{i-1} + K^{is} \circledast x_i)$$

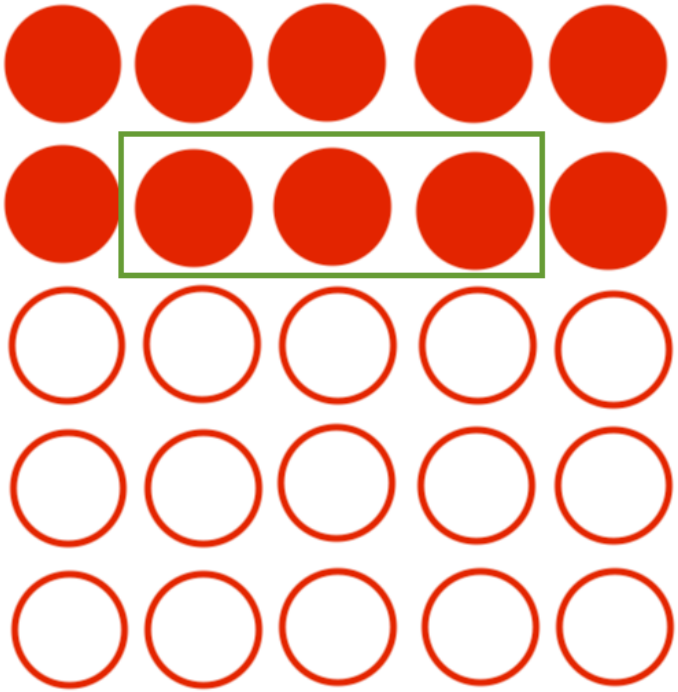


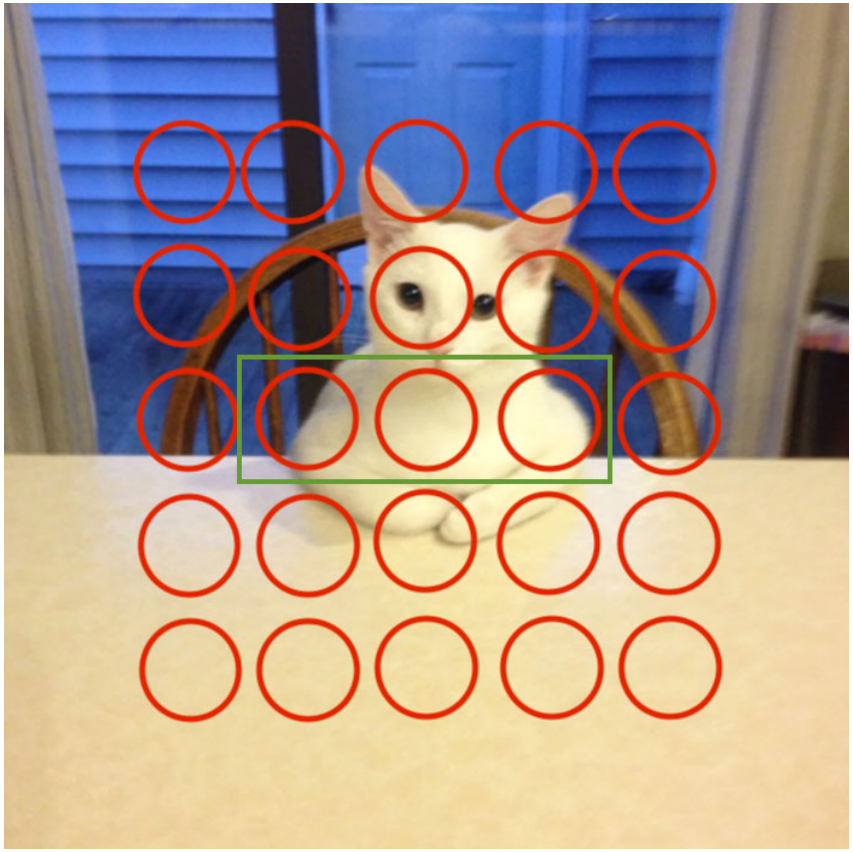


Input-to-state

Нужно посчитать:

$$\sigma(K^{ss} \circledast h_{i-1} + K^{is} \circledast x_i)$$

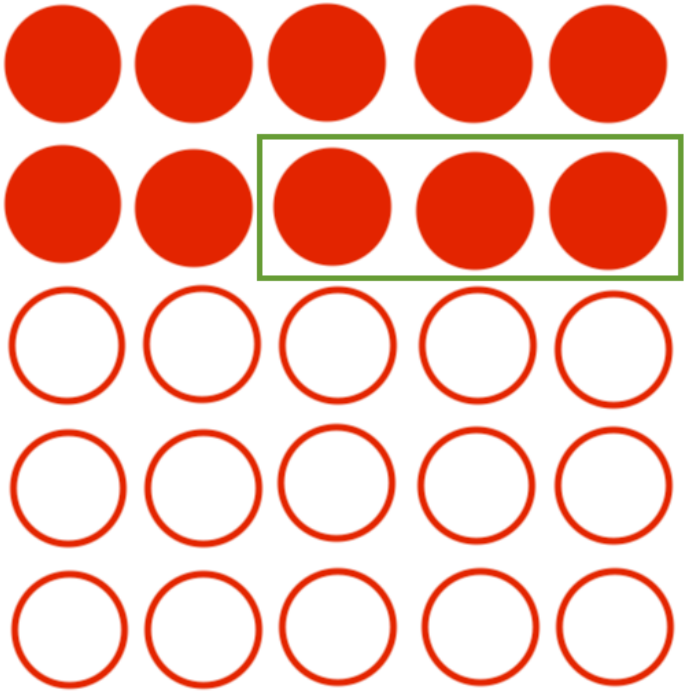


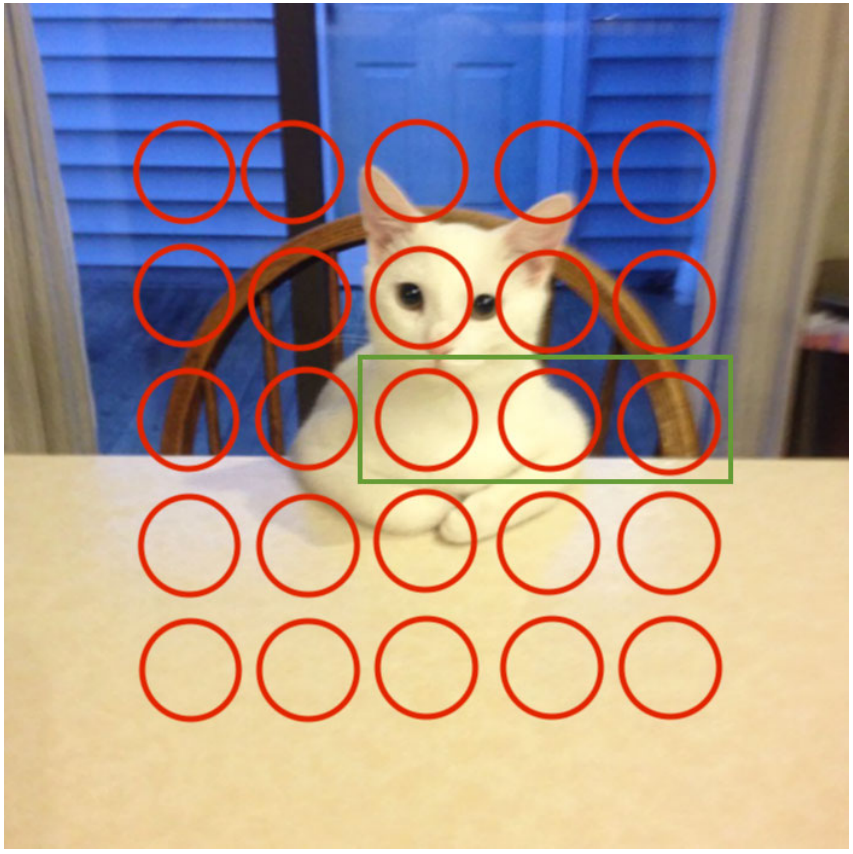


Input-to-state

Нужно посчитать:

$$\sigma(K^{ss} \circledast h_{i-1} + K^{is} \circledast x_i)$$

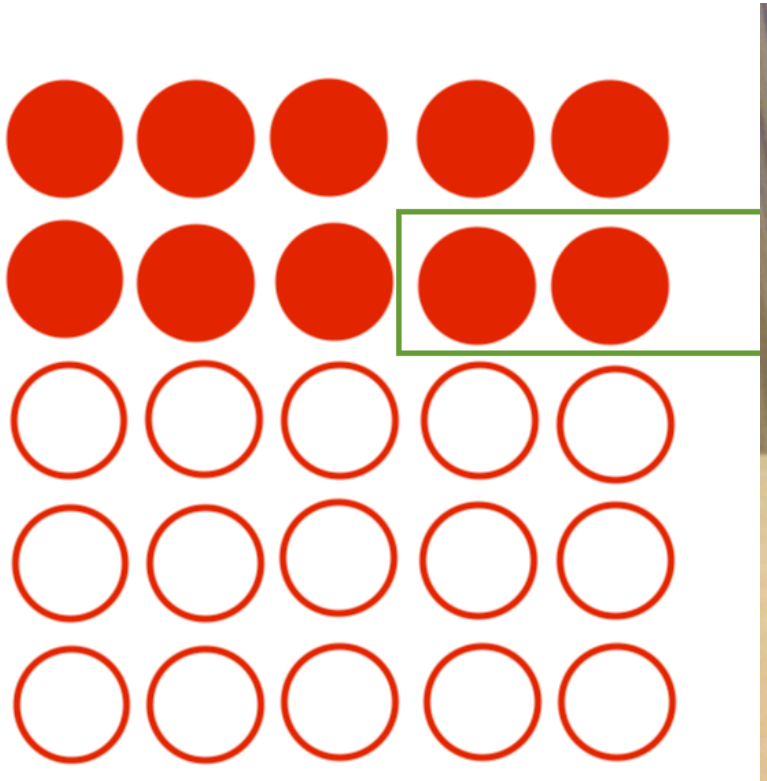


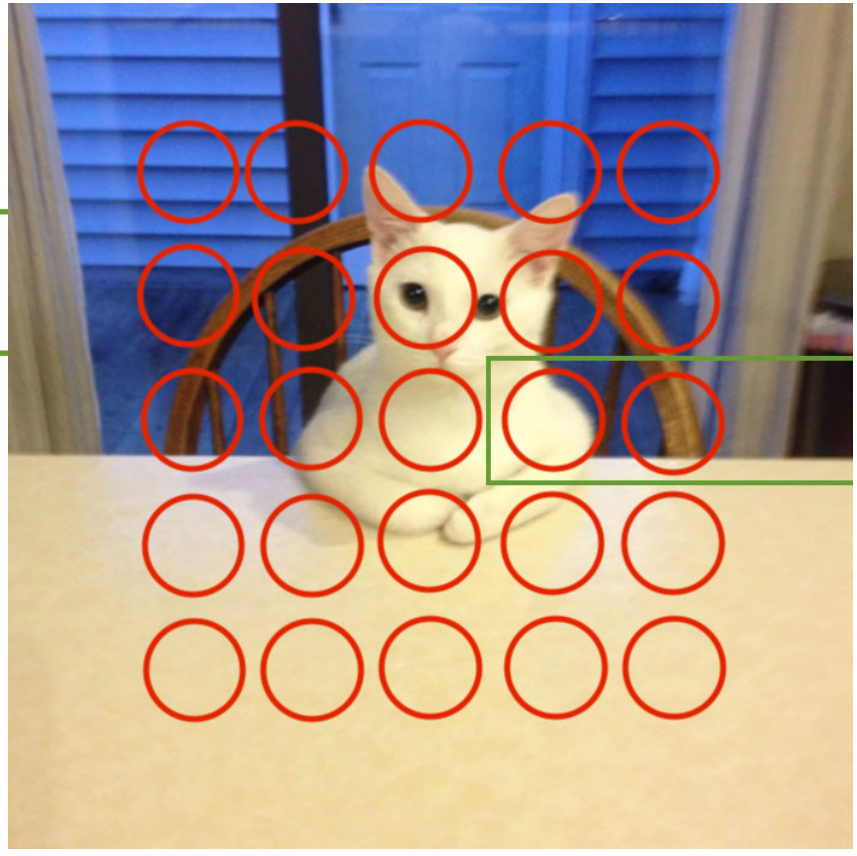


Input-to-state

Нужно посчитать:

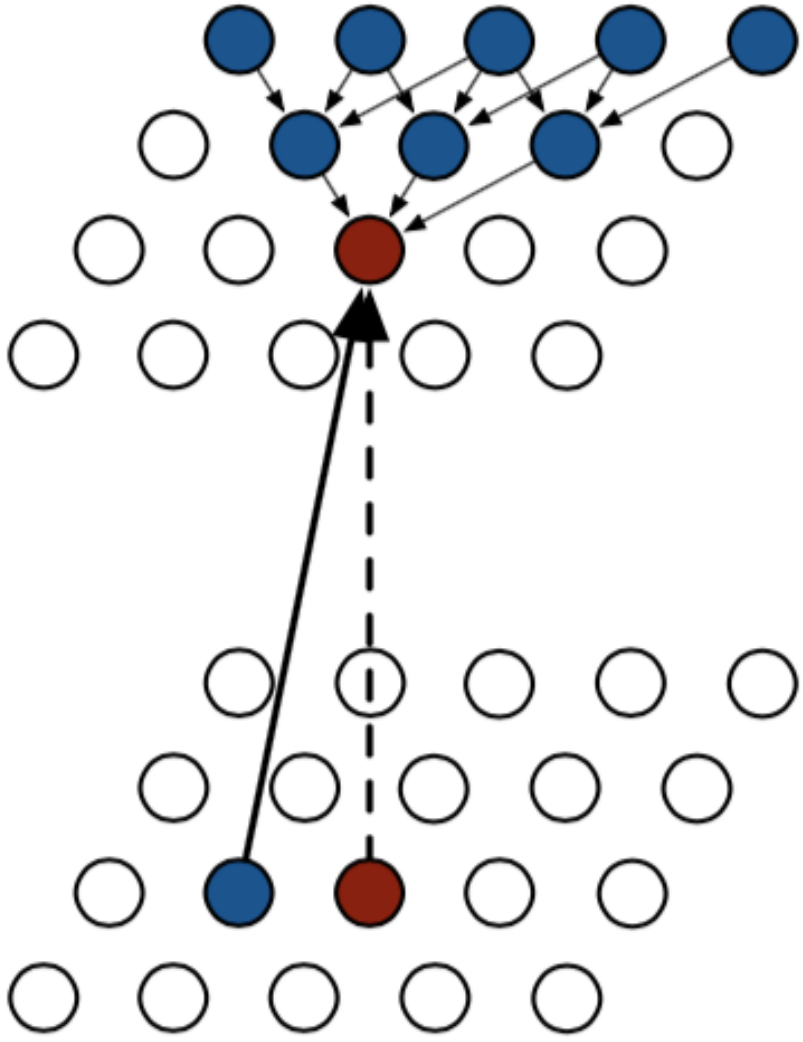
$$\sigma(K^{ss} \circledast h_{i-1} + K^{is} \circledast x_i)$$





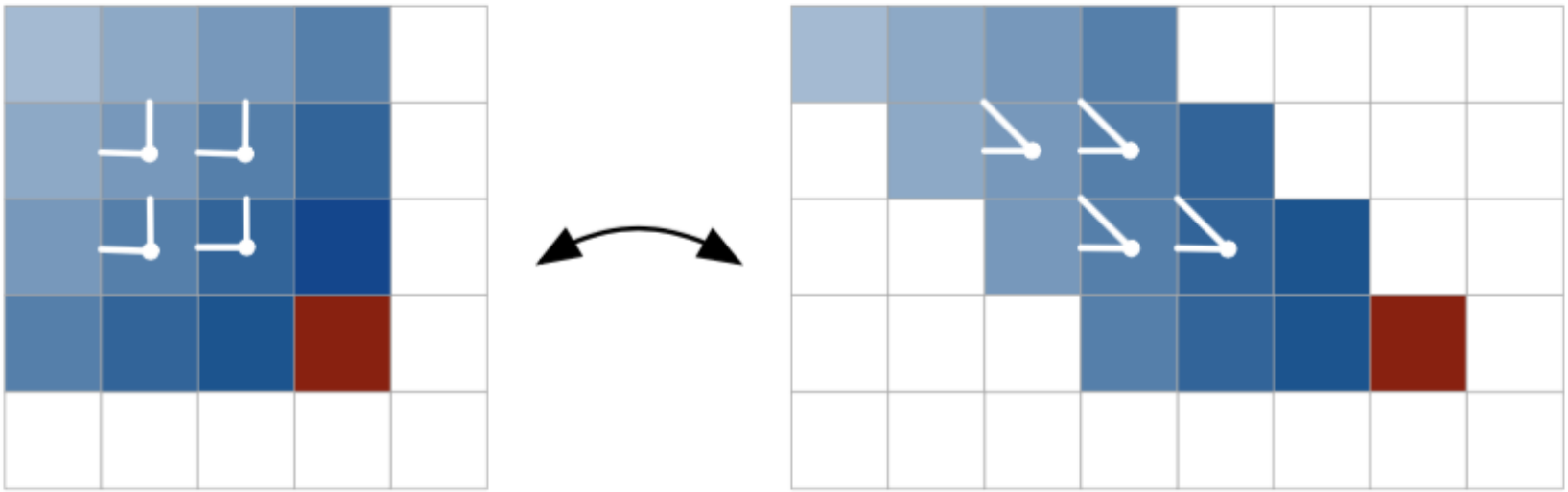
Недостаток Row LSTM

Используется не вся информация о предыдущих пикселях!

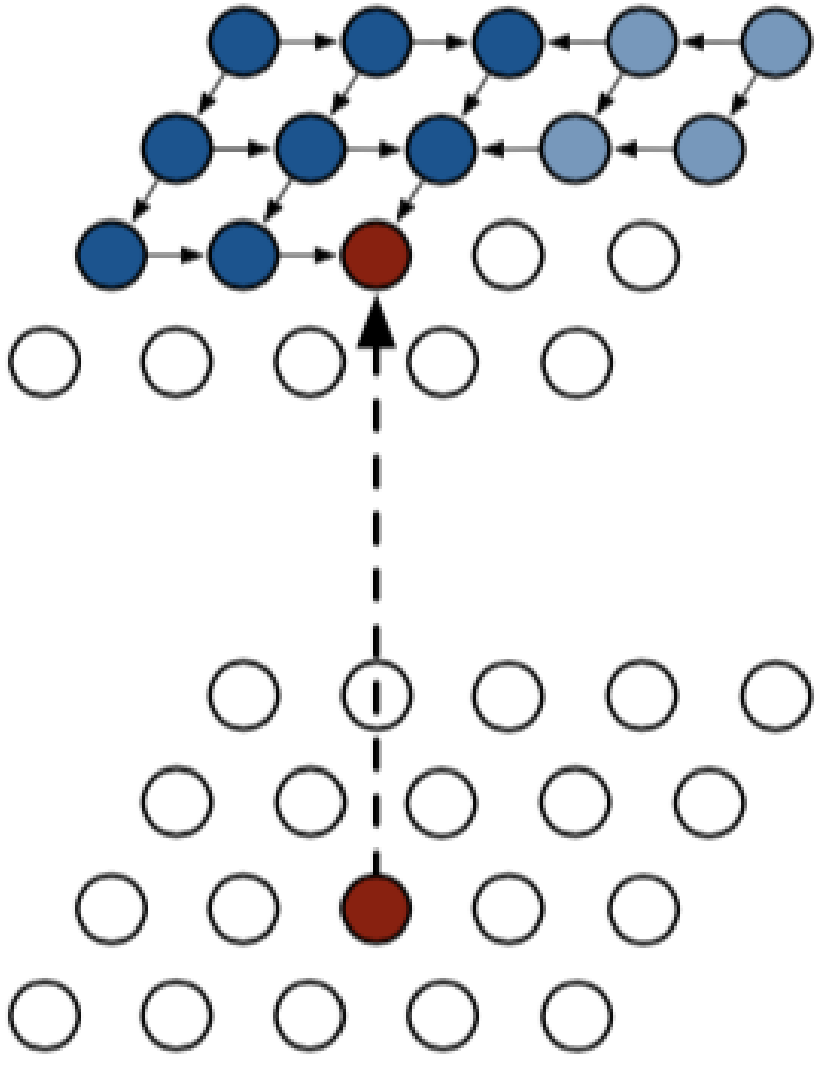


Diagonal BiLSTM

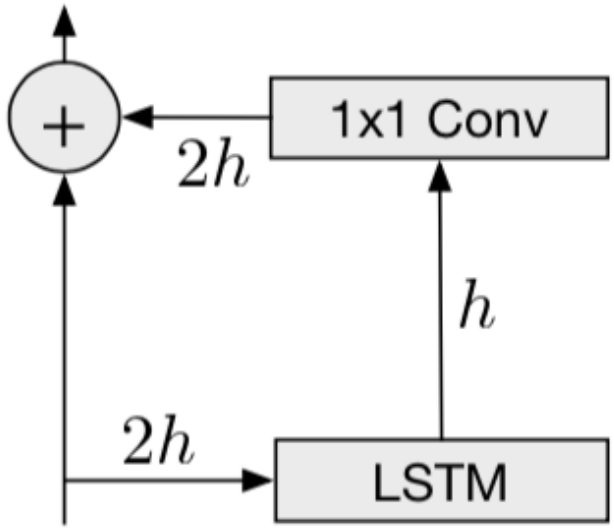
- Будем использовать не строки в качестве states, а диагонали.



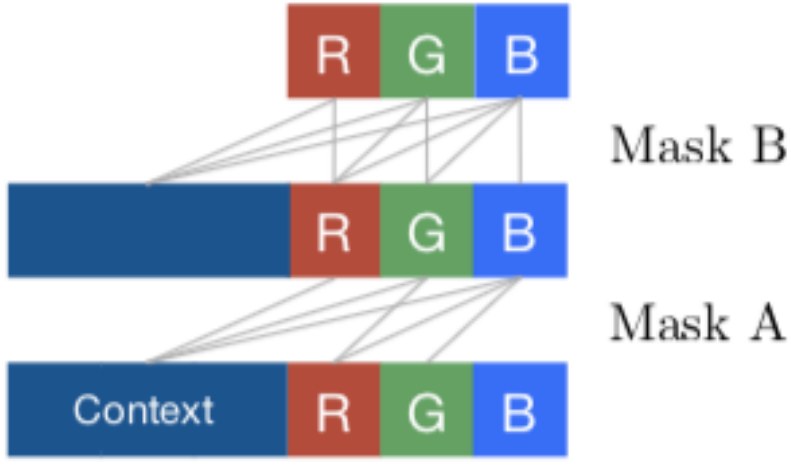
Diagonal BiLSTM



Residual connections



Masked Convolution



Pixel CNN

- Сделаем поле восприятия большим, но ограниченным.
- Используем сверточные слои
- Чтобы не захватывать будущие пиксели, к ядру свертки применим маску.
- Теперь можно распараллелить, т.к. удается посчитать все пиксели разом (не для задач генерации)

Specifications of models

PixelCNN	**Row LSTM**	**Diagonal BiLSTM**
7×7 conv mask A		
Multiple residual blocks: (see fig 5)		
Conv 3×3 mask B	Row LSTM i-s: 3×1 mask B s-s: 3×1 no mask	Diagonal BiLSTM i-s: 1×1 mask B s-s: 1×2 no mask
ReLU followed by 1×1 conv, mask B (2 layers)		
256-way Softmax for each RGB color (Natural images) or Sigmoid (MNIST)		

Residual connections

	No skip	Skip
No residual:	3.22	3.09
Residual:	3.07	3.06

Table 2. Effect of residual and skip connections in the Row LSTM network evaluated on the Cifar-10 validation set in bits/dim.

Cifar-10

Model	NLL Test (Train)
Uniform Distribution:	8.00
Multivariate Gaussian:	4.70
NICE [1]:	4.48
Deep Diffusion [2]:	4.20
Deep GMMs [3]:	4.00
RIDE [4]:	3.47
PixelCNN:	3.14 (3.08)
Row LSTM:	3.07 (3.00)
Diagonal BiLSTM:	**3.00** (2.93)

ImageNet

Image size	NLL Validation (Train)
32x32:	3.86 (3.83)
64x64:	3.63 (3.57)

Table 6. Negative log-likelihood performance on 32×32 and 64×64 ImageNet in *bits/dim*.

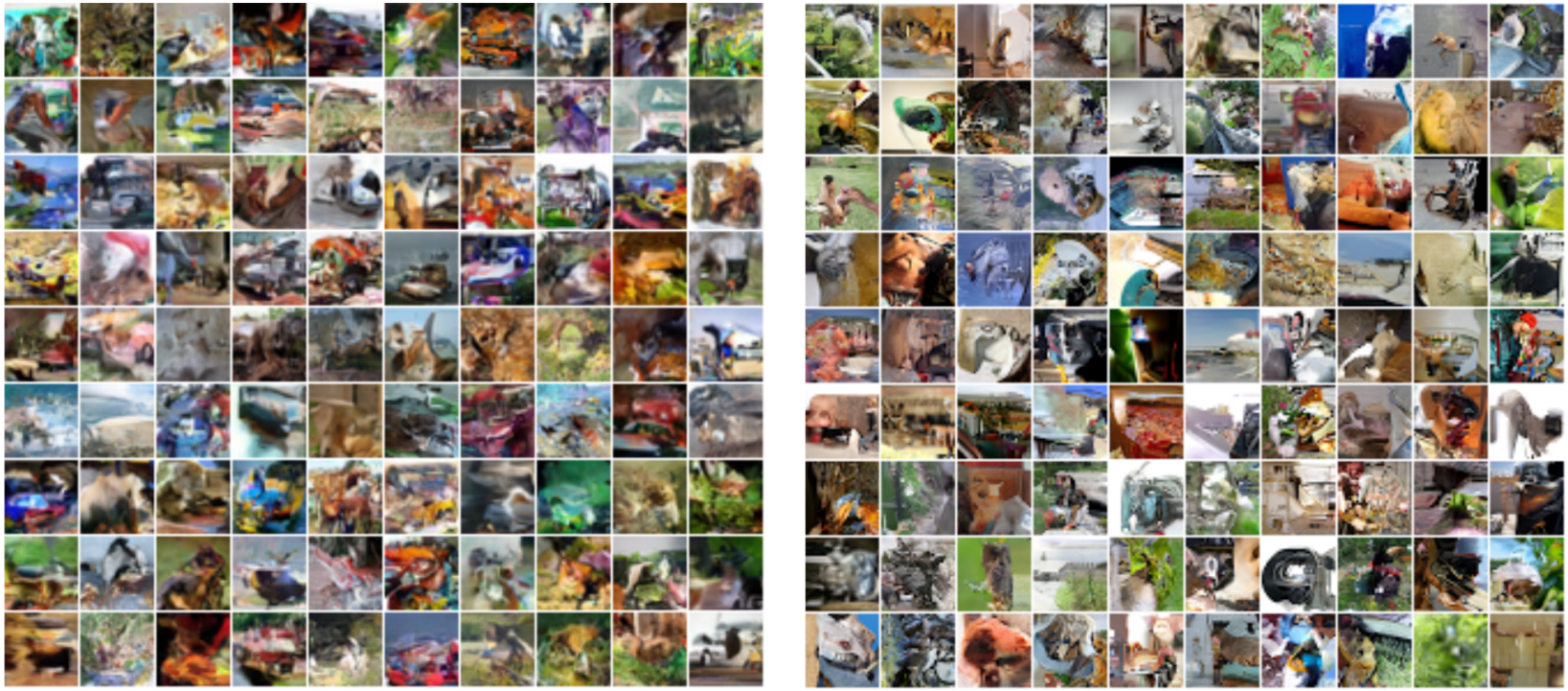
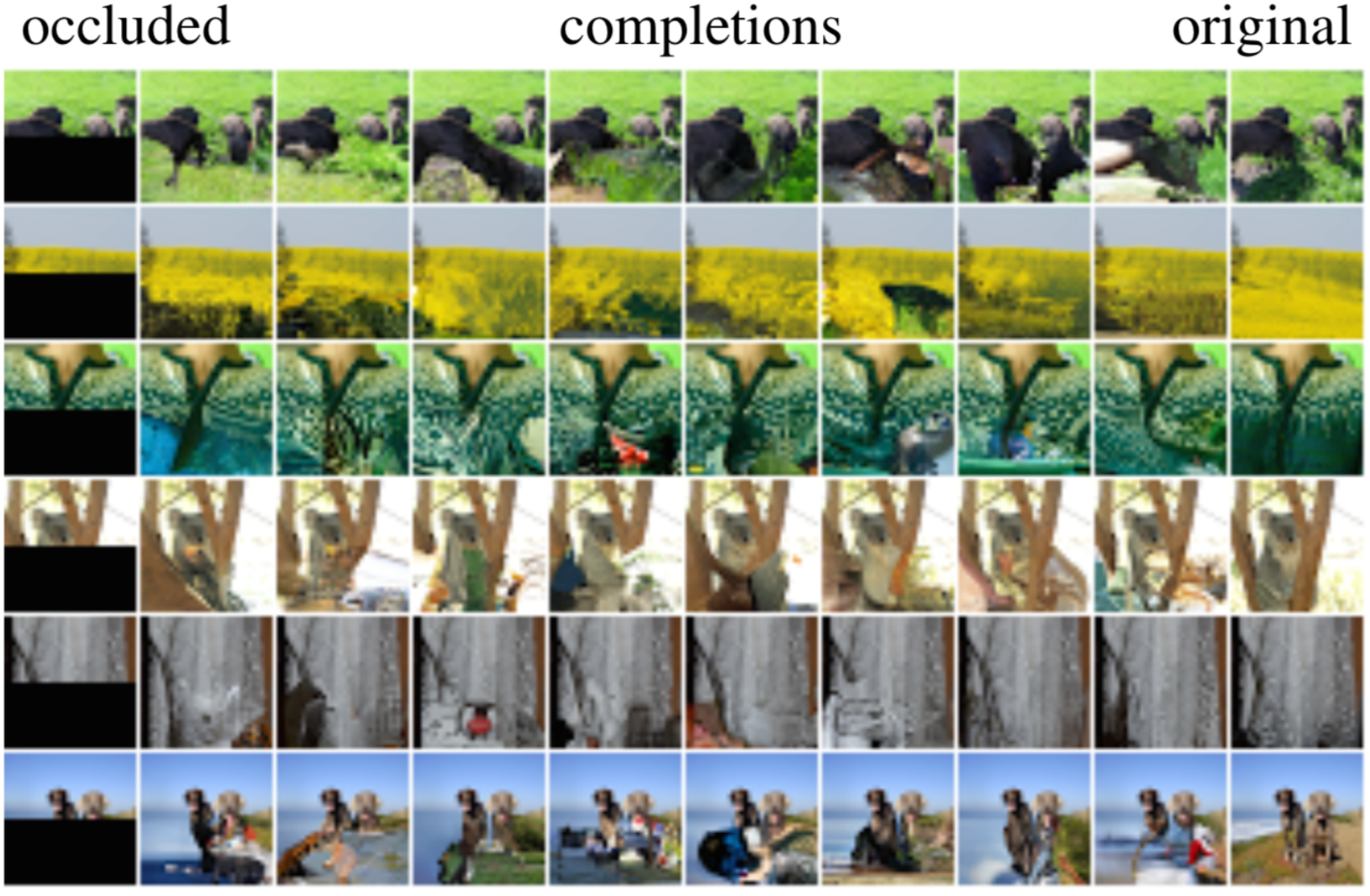


Figure 7. Samples from models trained on CIFAR-10 (left) and ImageNet 32x32 (right) images. In general we can see that the models capture local spatial dependencies relatively well. The ImageNet model seems to be better at capturing more global structures than the CIFAR-10 model. The ImageNet model was larger and trained on much more data, which explains the qualitative difference in samples.


occluded
completions
original

Literature

- https://arxiv.org/pdf/1601.06759.pdf
- http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- https://towardsdatascience.com/auto-regressive-generative-models-pixelrnn-pixelcnn-32d192911173