

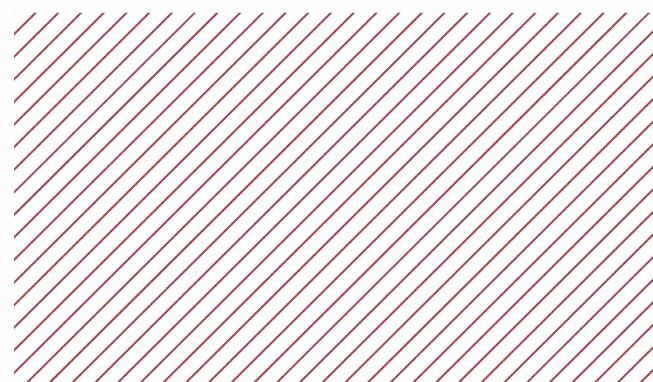
академия  
больших  
данных



# Введение в CV Аугментация изображений

Иван Карпухин

Ведущий программист-исследователь в  
команде машинного зрения



# Введение

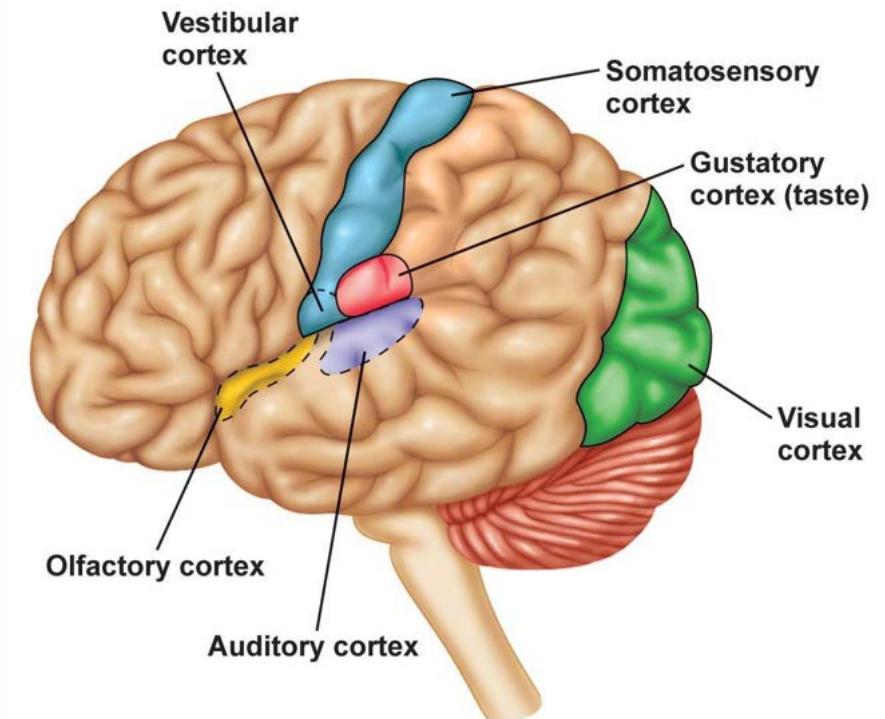


# Что изучает Computer Vision

Два подхода к определению CV:

**Биология:** повторить поведение зрительной коры головного мозга

**Инженерия:** автоматически извлекать высокоуровневую информацию из изображений и видео



© 2011 Pearson Education, Inc.

\* Huang T. Computer vision: Evolution and promise. 1996.

# История развития CV

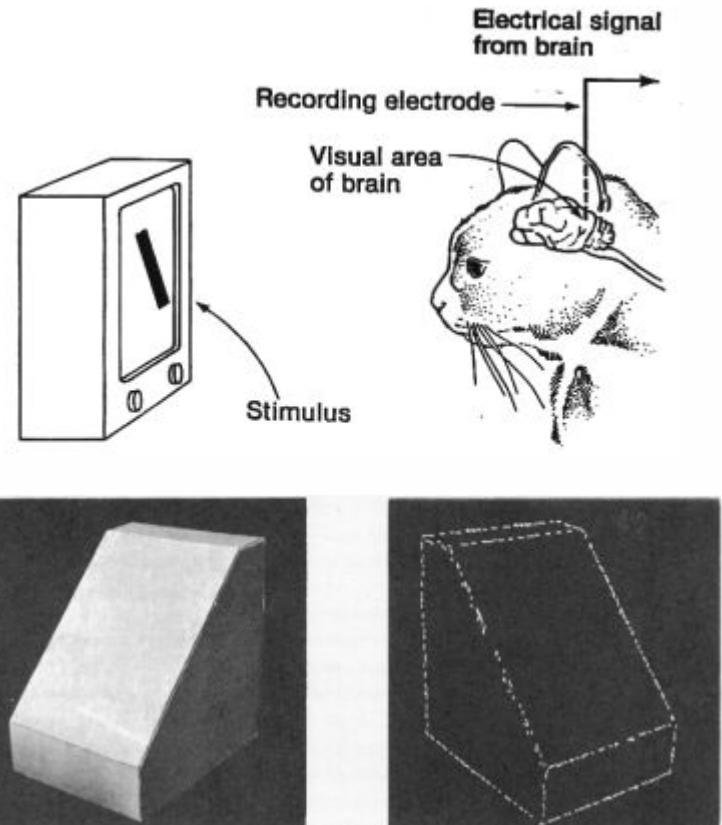
Появление: 1950–1970

1957: Цифровой сканер

1959: Изучение зрительной коры

1963: Реконструкция формы 3D моделей

1966: The Summer Vision Project



\* Hubel D. H., Wiesel T. N. Receptive fields of single neurones in the cat's striate cortex. 1959.

\* Roberts L. G. Machine perception of three-dimensional solids. 1963.

\* Papert S. A. The summer vision project. 1966.

# История развития CV

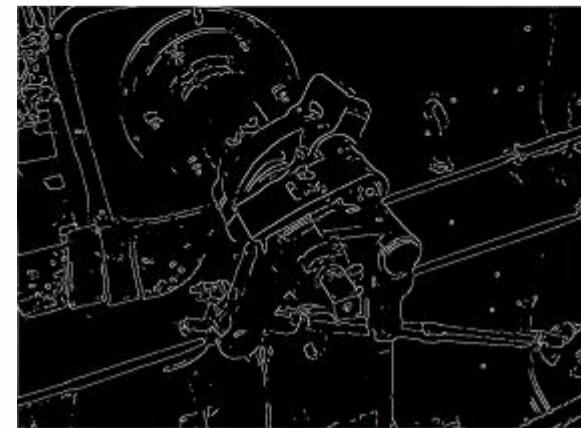
---

Классические алгоритмы CV: 1970–1990

1981: Оценка оптического потока (движения объектов на изображении)

1982: Теория анализа визуальной информации

1986: Детектор границ Кэнни



\* Horn B. K. P., Schunck B. G. Determining optical flow. 1981.

\* Marr D., Vision: A computational investigation into the human representation and processing of visual information. 1982.

\* Bao P., Zhang L., Wu X. Canny edge detection enhancement by scale multiplication. 2005.

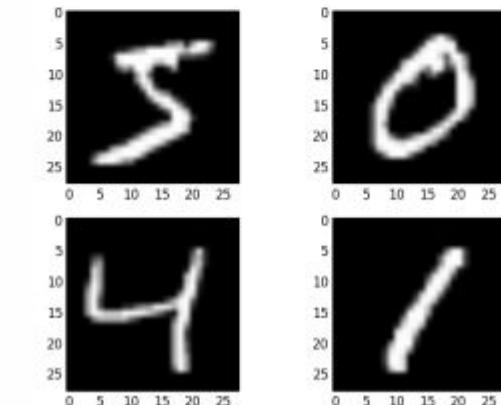
# История развития CV

---

Машинное обучение в CV: 1990–2000

1991: Распознавание лиц с использованием собственных векторов

1998: CNN для распознавания рукописных цифр



\* Turk M. A., Pentland A. P. Face recognition using eigenfaces. 1991.

\* LeCun Y. et al. Gradient-based learning applied to document recognition. 1998.

# История развития CV

---

Локальные дескрипторы: конец 2000–2010

1999: SIFT: локальные признаки, инвариантные к масштабу

2001: Метод Виолы — Джонса для поиска лиц

2006: SURF: быстро вычислимые локальные признаки

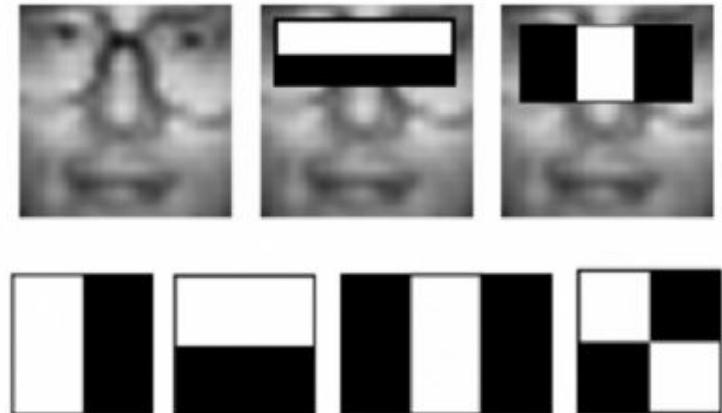
Области применения:

- 3D реконструкция
- поиск объектов
- распознавание лиц

\* Lowe D. G. Object recognition from local scale-invariant features. 1991.

\* Viola P. et al. Robust real-time object detection. 2001.

\* Bay H., Tuytelaars T., Van Gool L. Surf: Speeded up robust features. 2006.



# История развития CV

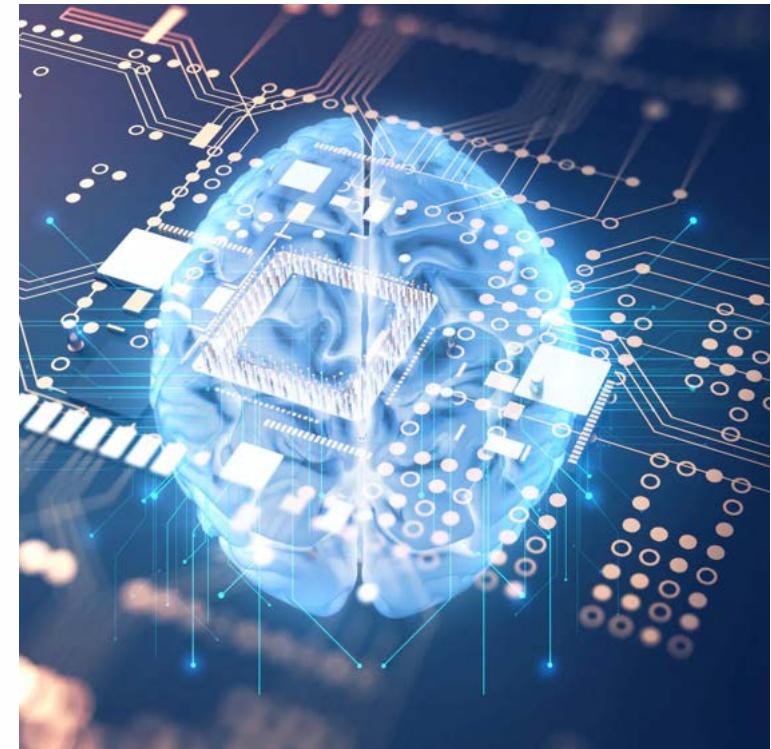
---

Глубокое обучение: с 2010

2012: AlexNet: прорыв в распознавании изображений из ImageNet

2014: VGG: больше слоев, выше качество

2015: ResNet: изменение архитектуры, качество выше, чем у человека



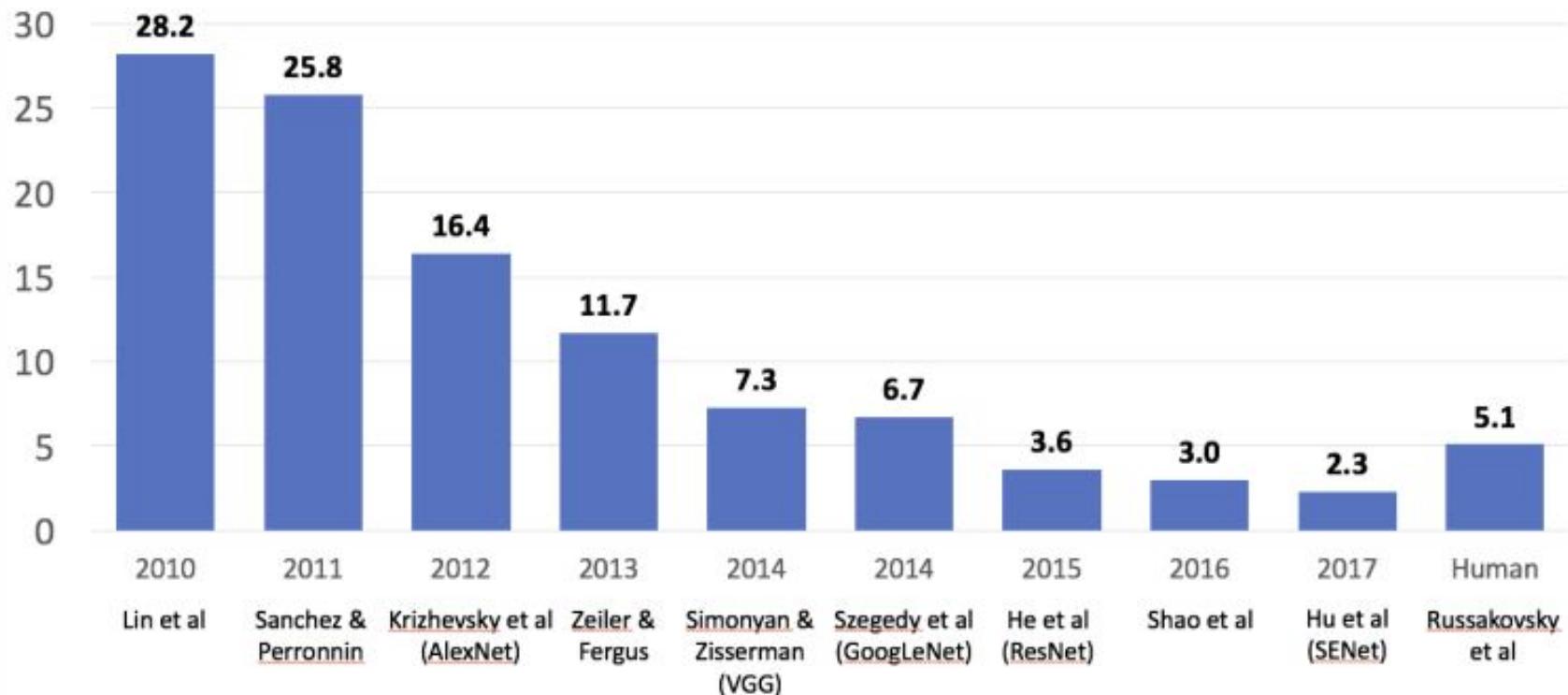
\* Krizhevsky A., Sutskever I., Hinton G. E. Imagenet classification with deep convolutional neural networks. 2012.

\* Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014.

\* He K. et al. Deep residual learning for image recognition. 2015

# История развития CV

Глубокое обучение: с 2010





# Вопросы

---

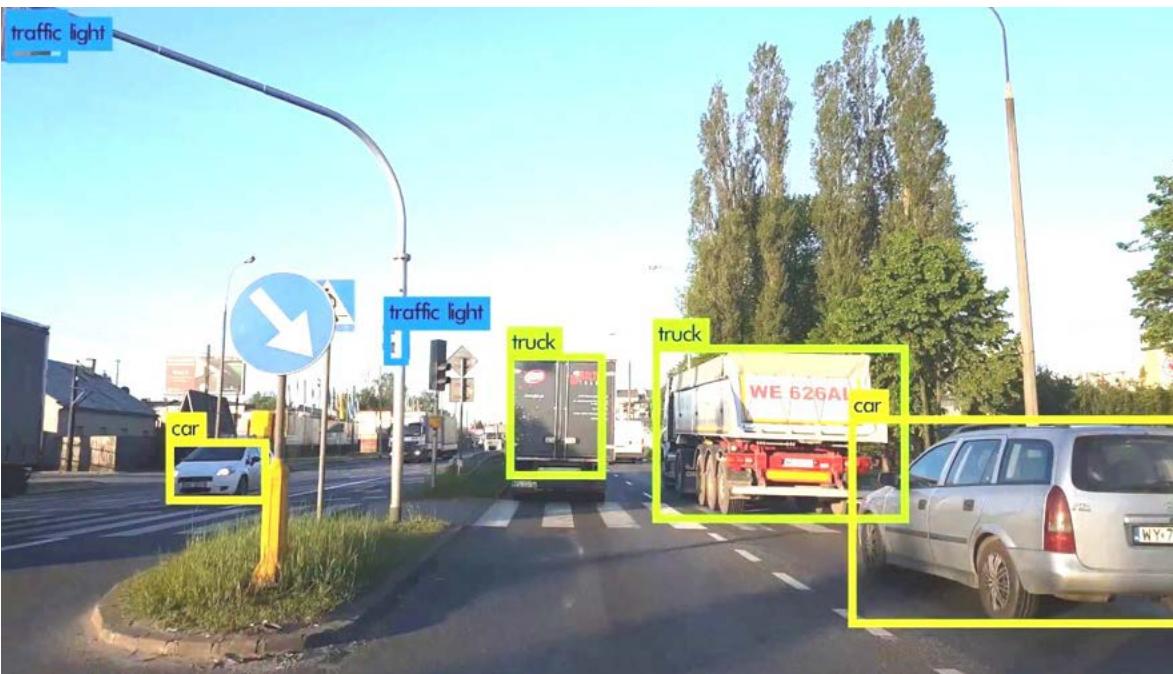
Kypc



# О чём этот курс

---

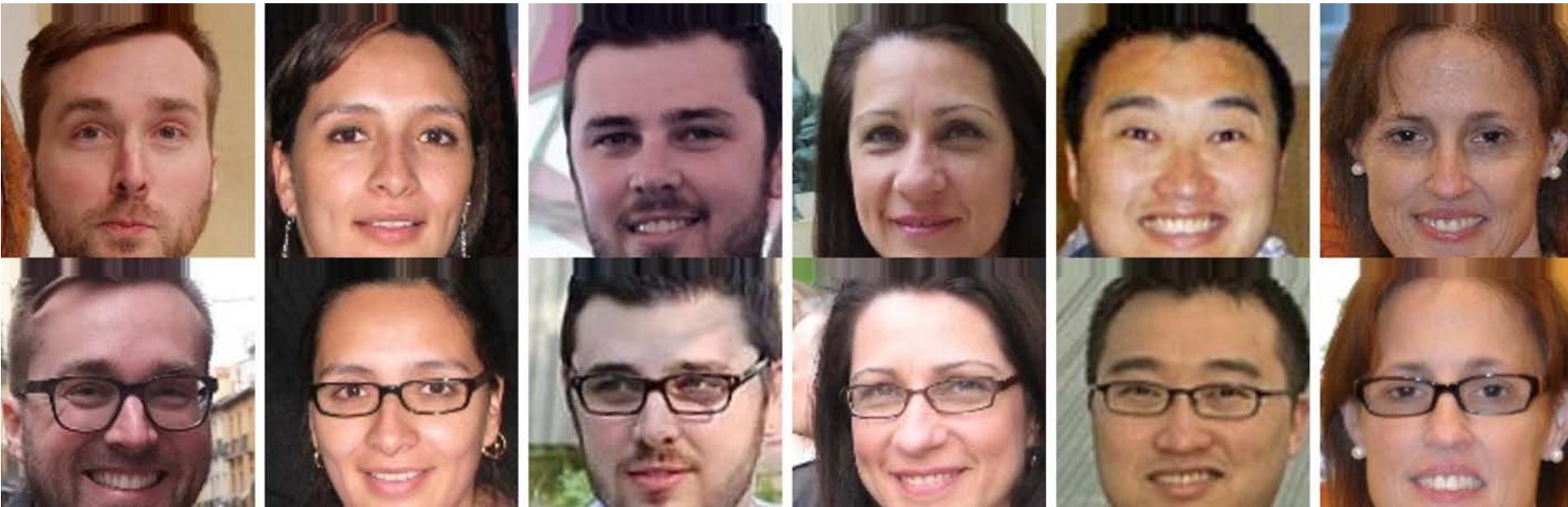
Современные задачи CV: поиск и распознавание объектов



# О чём этот курс

---

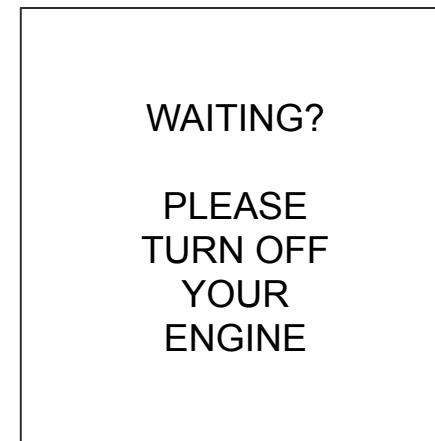
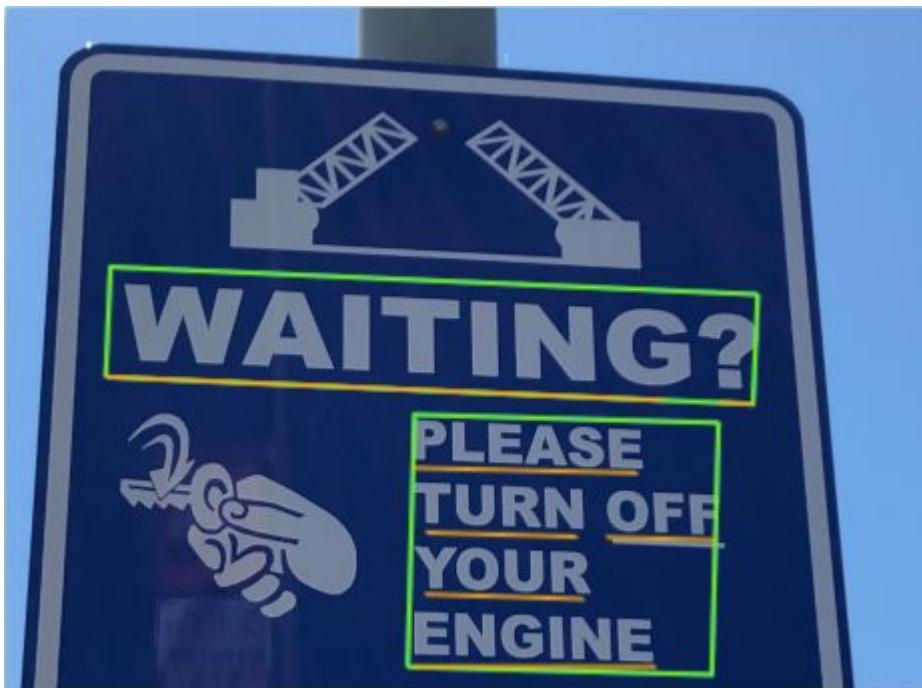
Современные задачи CV: распознавание лиц



# О чём этот курс

---

Современные задачи CV: распознавание текстов



# О чём этот курс

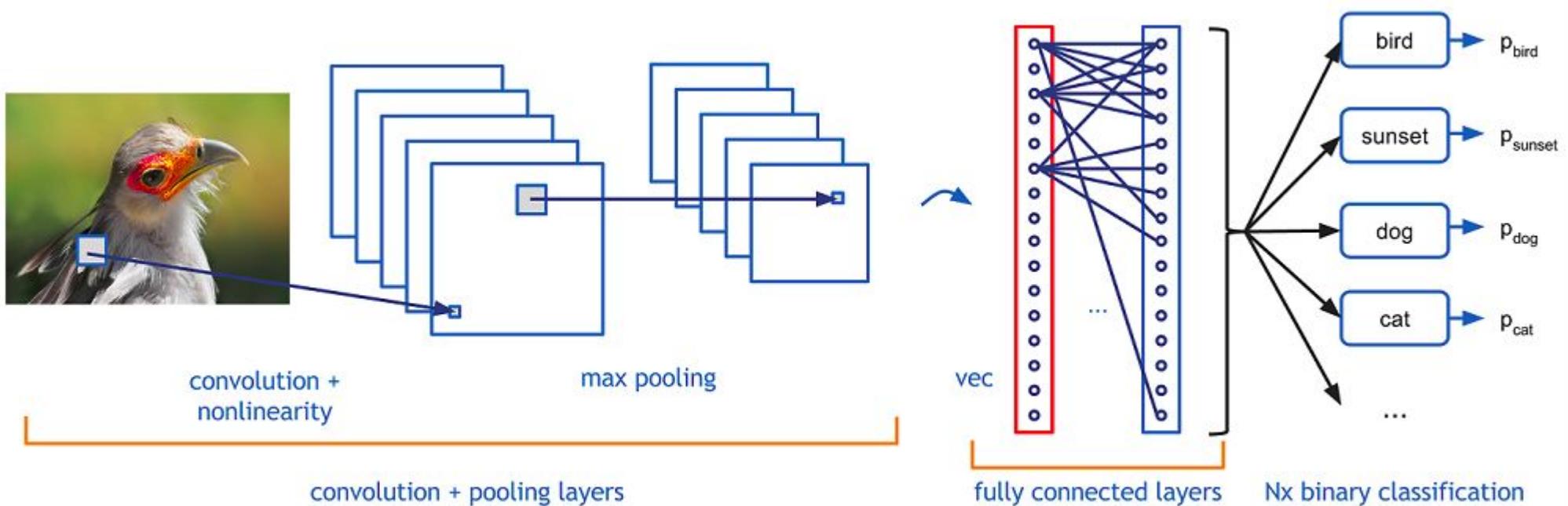
---

Современные задачи CV: генерация изображений



# О чём этот курс

Решение CV задач при помощи нейронных сетей  
(преимущественно свёрточных)

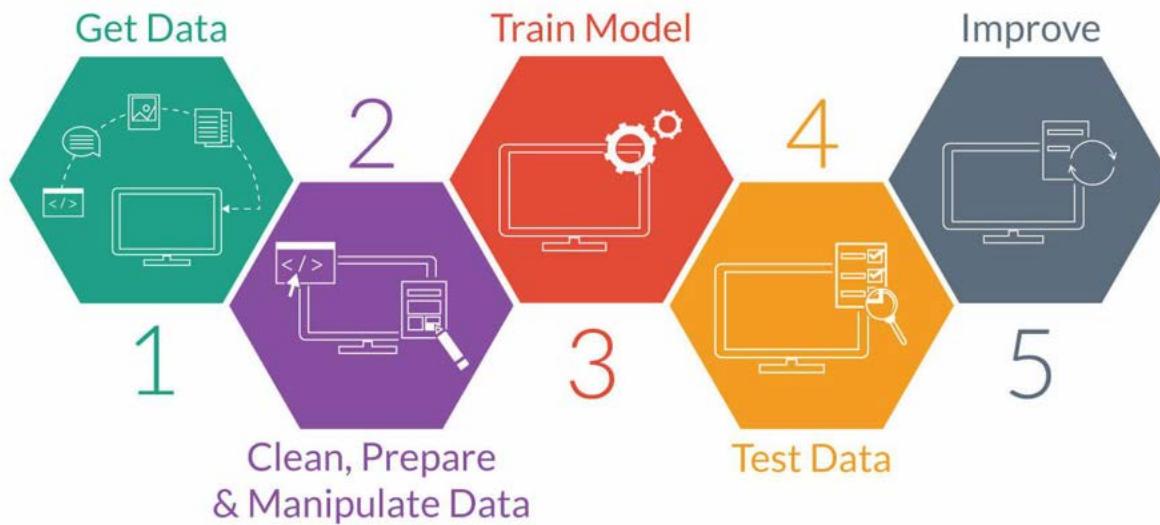


# О чём этот курс

---

Хотим научить:

- Решать современные задачи СV методами глубокого обучения
- Устранять возникающие проблемы
- Понимать специфику использования нейронных сетей в production





# Как устроен курс

---

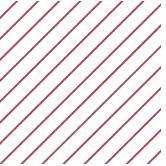
Начало: 4 марта

Конец: 3 июня (может измениться)

Каждый четверг: лекция и семинар

~13 занятий

2-3 контеста



# Как устроен курс

---

Темы:

- виды свёрточных сетей
- поиск объектов на изображениях
- распознавание объектов
- сегментация
- распознавание текстов
- генерация изображений
- особенности production систем

# Преподаватели

---



Андрей Бояров

Ведущий программист-исследователь  
в команде машинного зрения



Даниил Лысухин

Программист-исследователь  
в команде машинного зрения



Фёдор Киташов

Программист-исследователь  
в команде машинного зрения



Иван Карпухин

Ведущий программист-исследователь  
в команде машинного зрения



Борис Лесцов

Программист-исследователь  
в команде машинного зрения



Юлиана Журавлёва

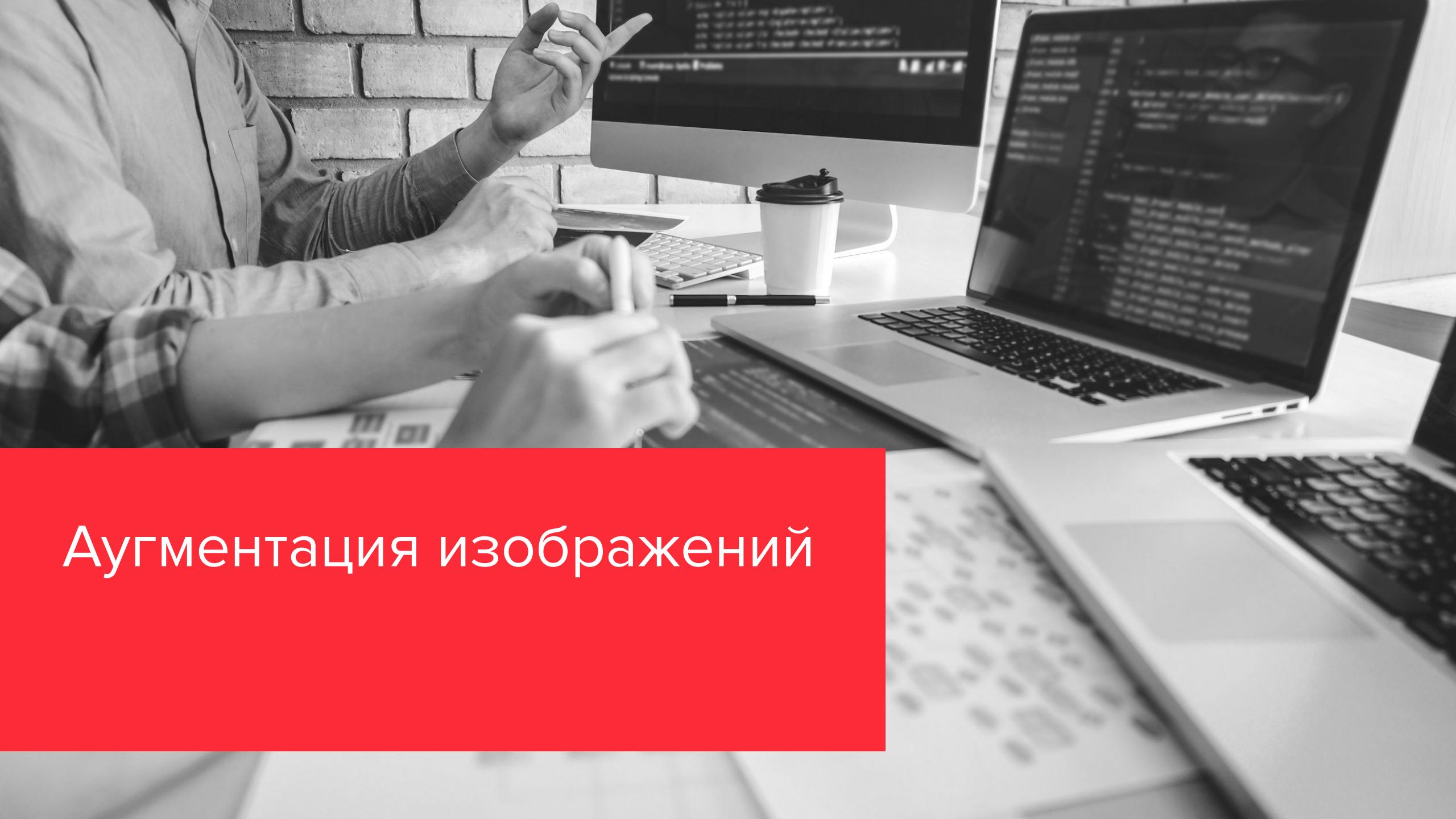
Руководитель группы С/С++  
разработки



# Вопросы

---

# Аугментация изображений

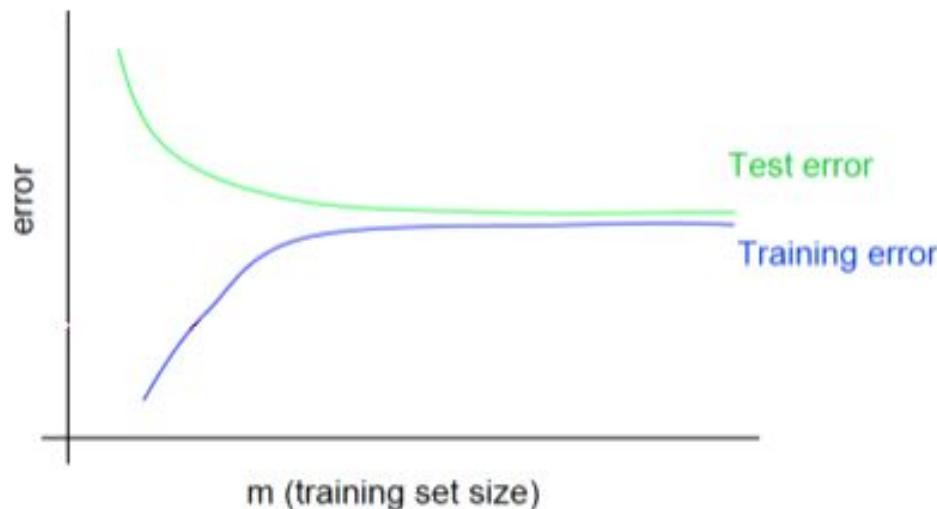


# Цель аугментации

---

Теория:

- Для обучения сетей нужно много данных
- Тренировочный корпус должен быть близким по распределению к тестовому



# Цель аугментации

---

Теория:

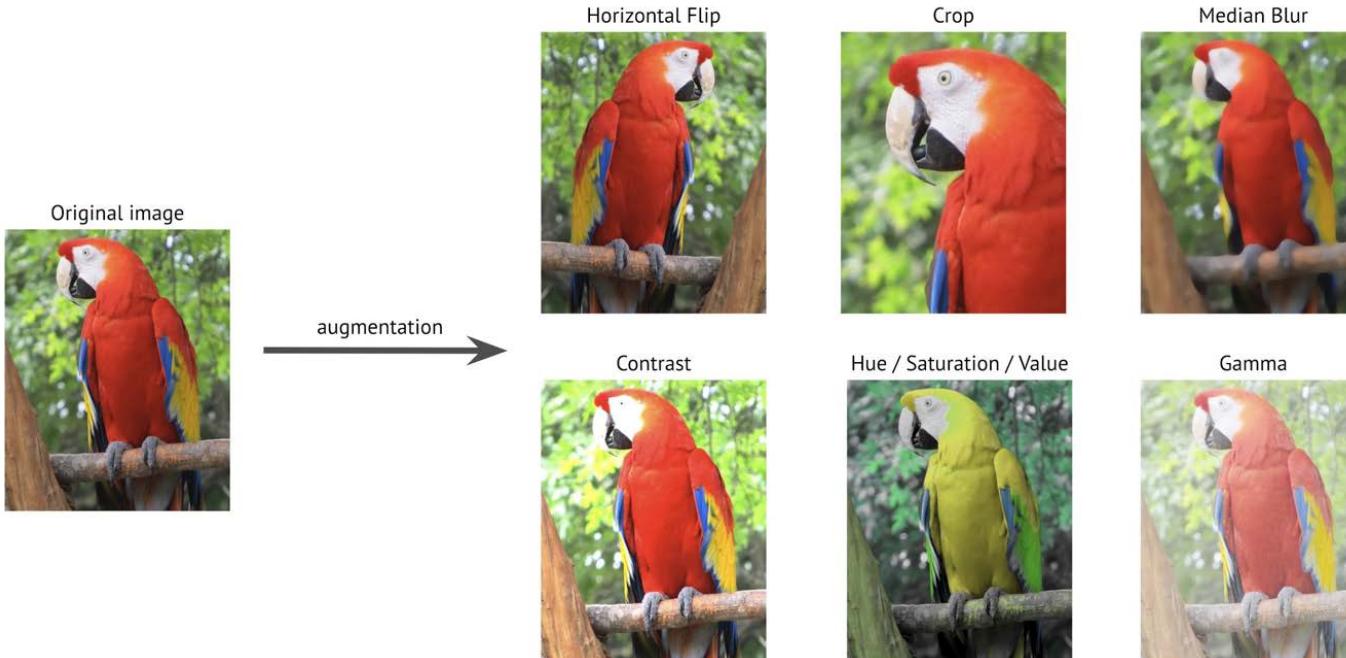
- Для обучения сетей нужно много данных
- Тренировочный корпус должен быть близким по распределению к тестовому

На практике достаточно, чтобы в `train` было много примеров из `test` распределения. Близкие, но отличные от `test` примеры также допустимы. Сеть научится работать в разных доменах. Плюс, станет лучше обобщать.

# Аугментация

Аугментация - добавление измененных примеров в тренировочный корпус

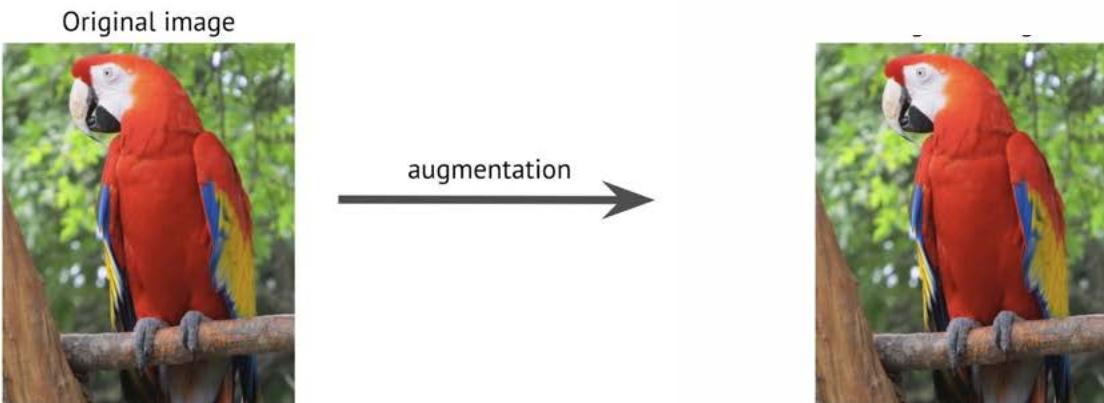
- увеличивает размер train
- повышает разнообразие данных
- играет роль регуляризации



# Замечания

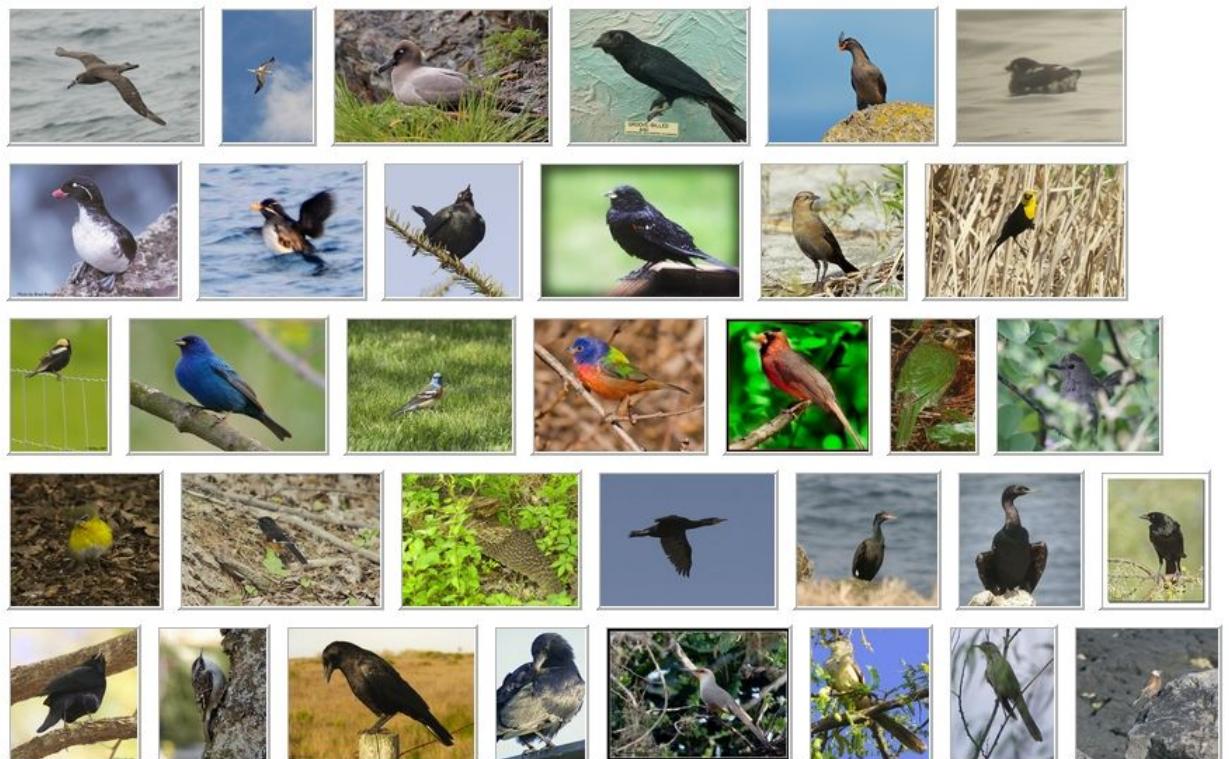
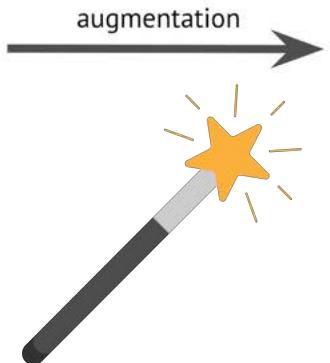
---

Тривиальная аугментация:



# Замечания

Идеальная аугментация:

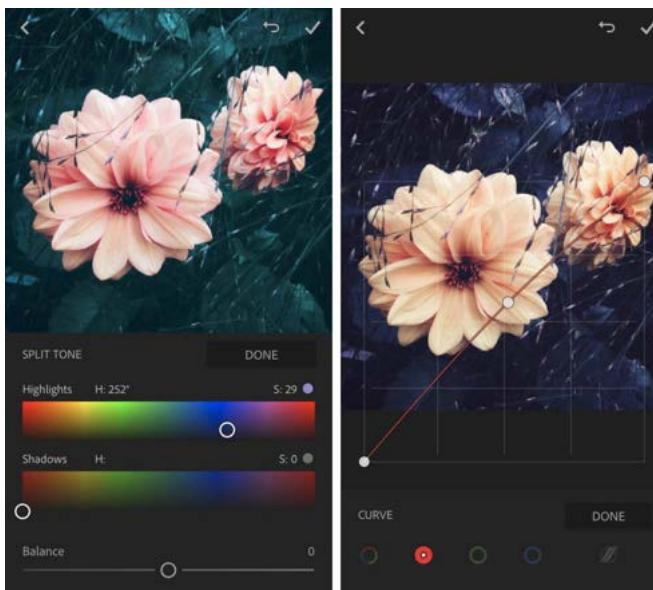
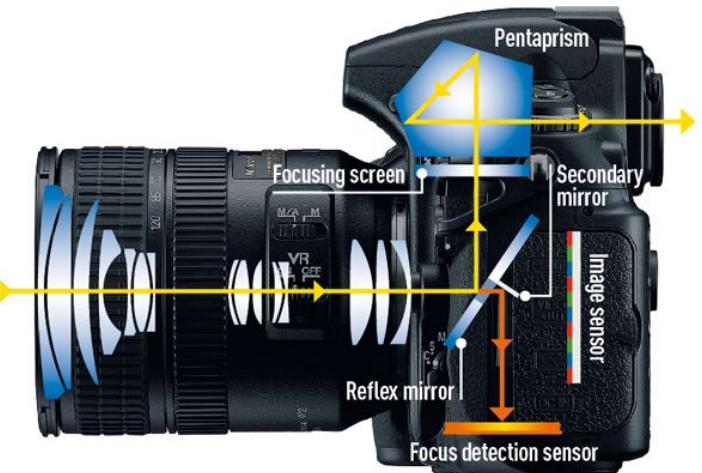


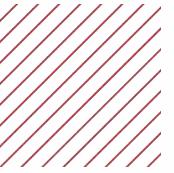
Full dataset

# Замечания

Аугментация - способ внести априорное знание в процесс обучения:

- источник данных
- способ обработки данных перед попаданием в систему
- особенности зрительного восприятия человека





# Вопросы

---

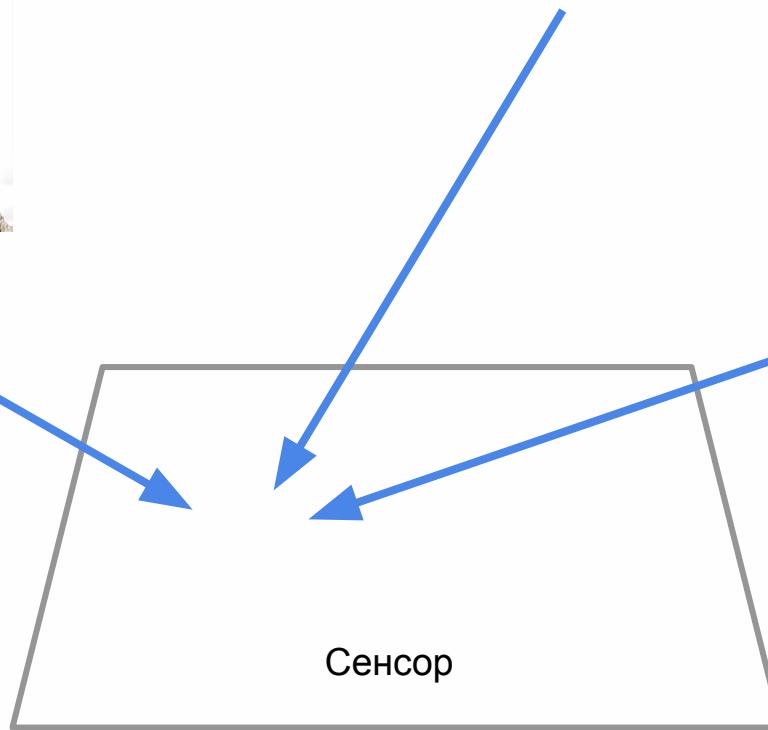
```
background-color: #F5F5F5;
text-shadow: 0px -1px 0px #EAEAEA;
filter: dropshadow(color:#777;
color:#777;

}
header #main-navigation ul li span:hover,
box-shadow: 0px 0px 1px #EAEAEA;
-webkit-box-shadow: 0px 0px 2px #EAEAEA;
moz-box-shadow: 0px 0px 1px #EAEAEA;
background-color:#F9F9F9;
active span;
li span.dashboard,
..../img/dashboard.png);
ul li span.dashboard,
finaldashboard.html);
background-color: #F5F5F5;
background-color: #F5F5F5;
```

## Оптические искажения

# Оптика

---



# Оптика: камера-обскура (стеноп)

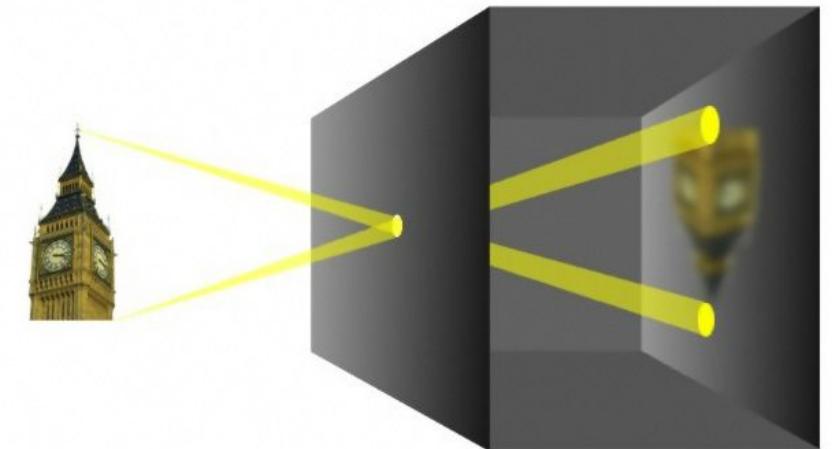
---

Отверстие увеличивается:

- увеличивается яркость изображения
- уменьшается резкость

(и наоборот)

Объекты на любом расстоянии в фокусе.



# Оптика: камера-обскура (стеноп)

---

Выдержка 20 минут

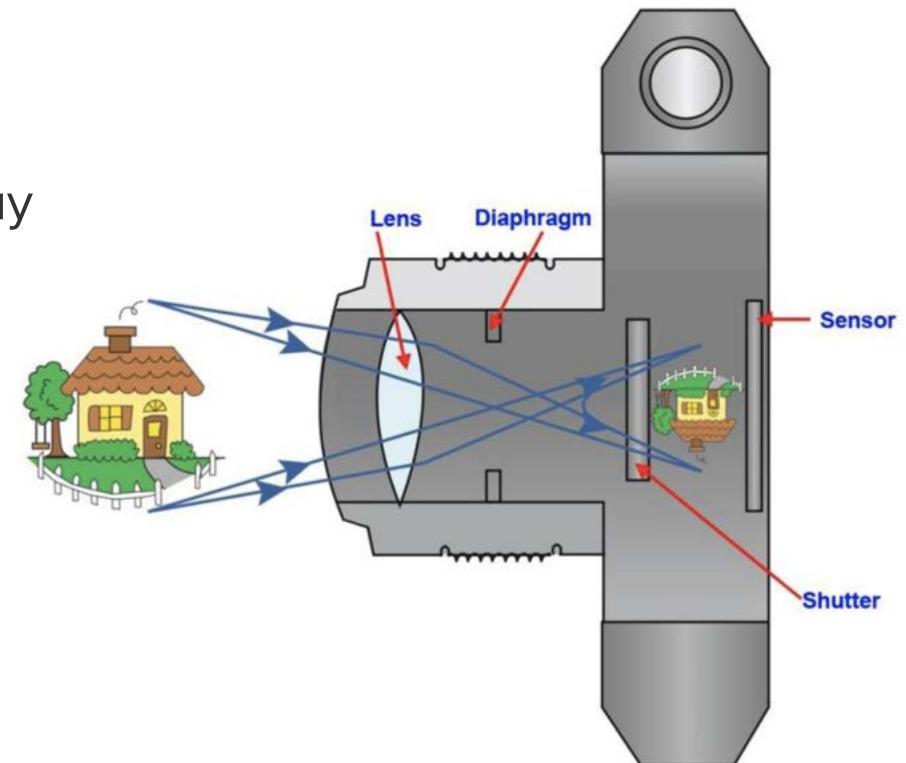


# Оптика: фотокамера

Добавляются линзы (объектив)

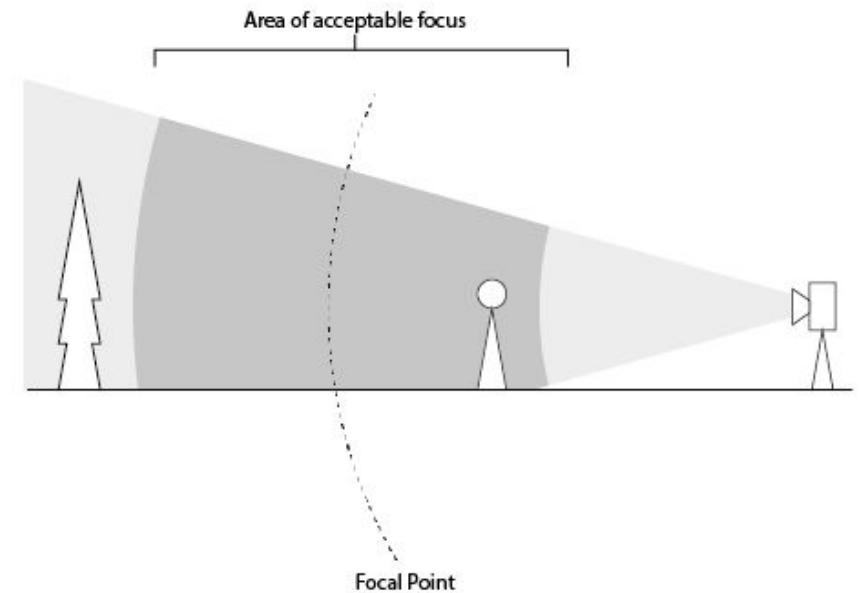
Роль отверстия стенопа выполняет диафрагма

- + Линза фокусирует изображение
- + Можно использовать более широкую диафрагму
- + Нужна меньшая выдержка
- - Часть объектов “в фокусе”, часть нет
- - Появляются оптические искажения



# Оптика: резкость

Ширина диафрагмы и фокус влияют на резкость



# Оптика: яркость

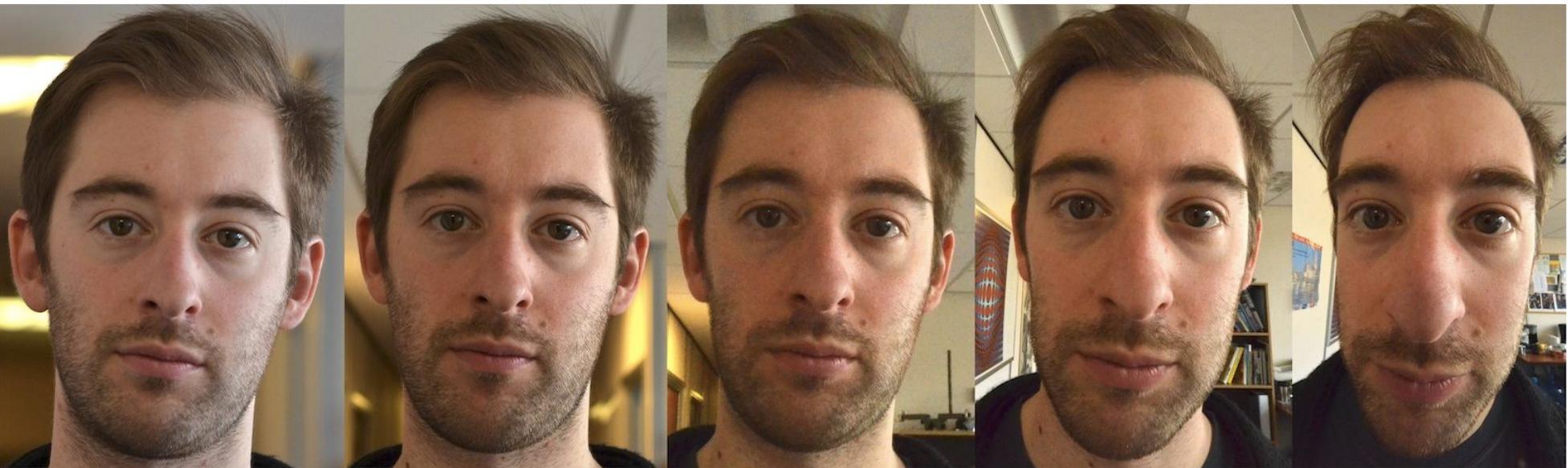
---

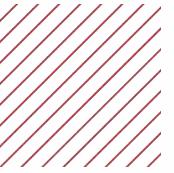
Ширина диафрагмы и выдержка влияют на яркость



# Оптические искажения

---



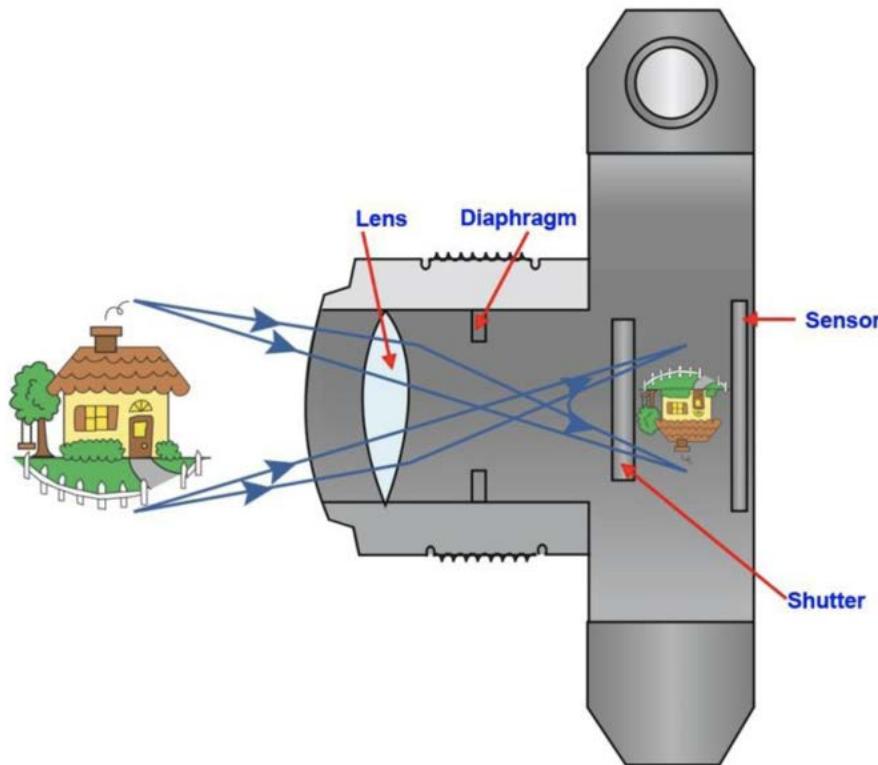


# Вопросы

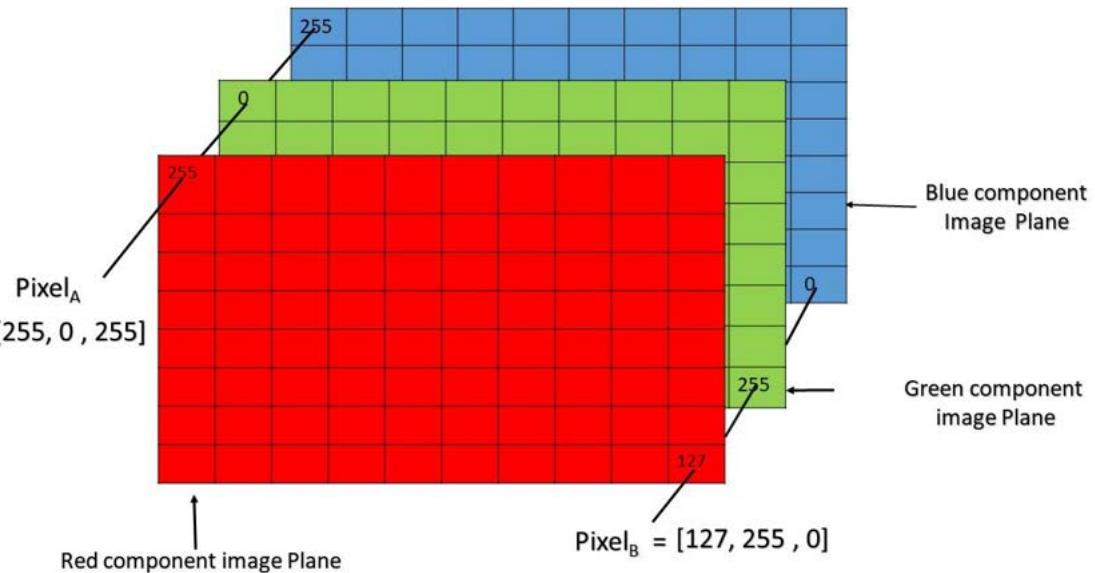
---

# Кодирование изображений

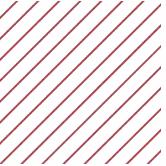
---



# Кодирование изображений



Pixel of an RGB image are formed from the corresponding pixel of the three component images



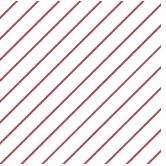
# Кодирование яркости

---

В простейшем случае кодируем яркость пикселя Float16 или Float32

Размер 4K изображения во Float32 без сжатия:

$$4096 * 3072 * 3 * 4 \text{ байт} = 144 \text{ Мбайт}$$



# Кодирование яркости

---

В простейшем случае кодируем яркость пикселя Float16 или Float32

Размер 4К изображения во Float32 без сжатия:

$$4096 * 3072 * 3 * 4 \text{ байт} = 144 \text{ Мбайт}$$

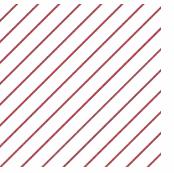
Чаще используют unsigned int 8 с диапазоном [0, 255]

Следовательно: потеря информации

# Потеря информации

---

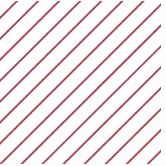




# Вопросы

---

## Примеры аугментаций



# Виды аугментаций

---

- Цветовые
- Размытие
- Зашумление
- Геометрические аффинные
- Прочие геометрические

# Цвет: яркость и контраст

---

x - исходная яркость (R, G или B)

c - множитель контраста в диапазоне [0, 1]

b - изменение яркости в диапазоне [0, 255]

$$f(x) = c(x - 128) + 128 + b$$

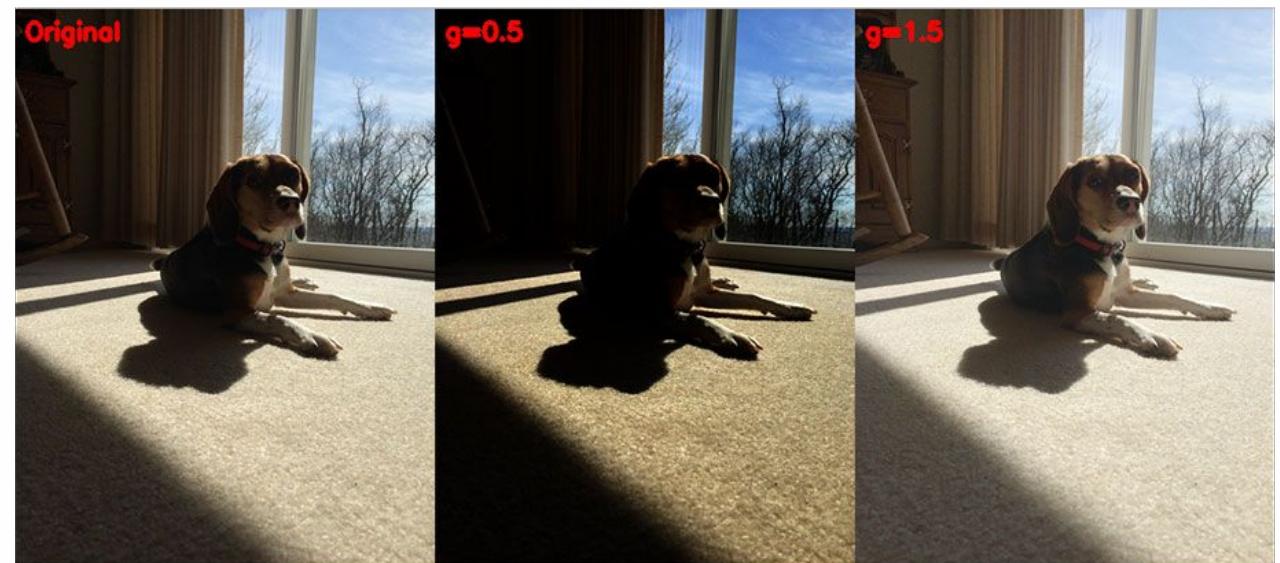
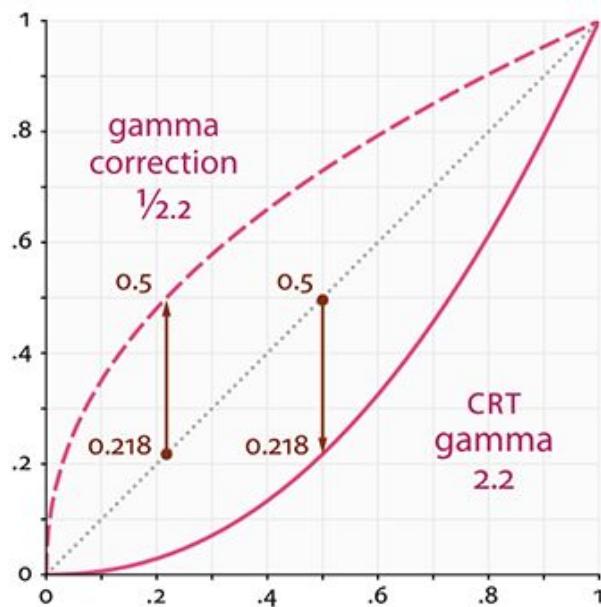
(линейное преобразование яркости)



(!) Нужно помнить о возможном переполнении uint8 данных

# Цвет: гамма-коррекция

- Нелинейное преобразование
- Не происходит переполнения  
(черный переходит в черный, белый в белый)



# Цвет: изменение оттенка и насыщенности

---

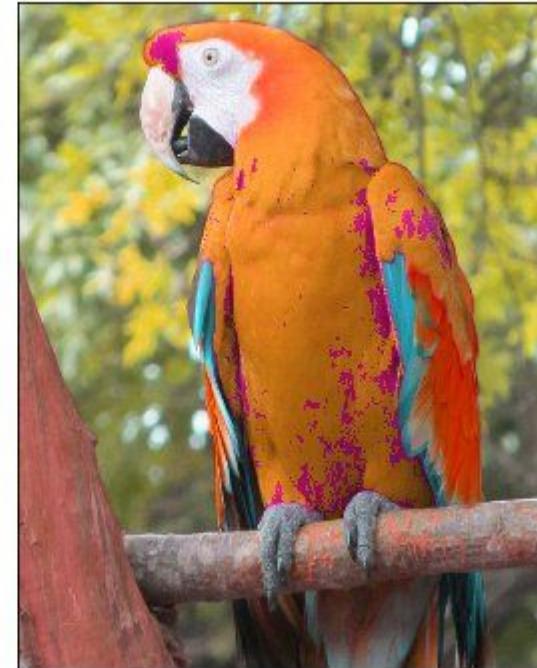
Original image



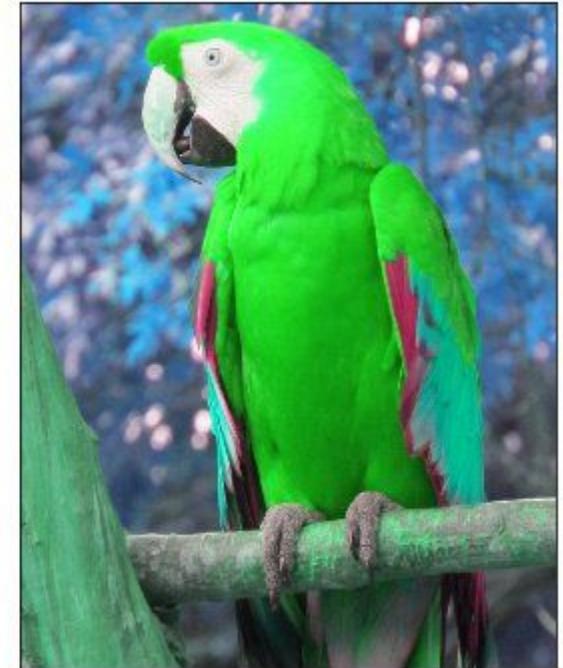
RGBShift



HueSaturationValue



ChannelShuffle



# Зашумление

---

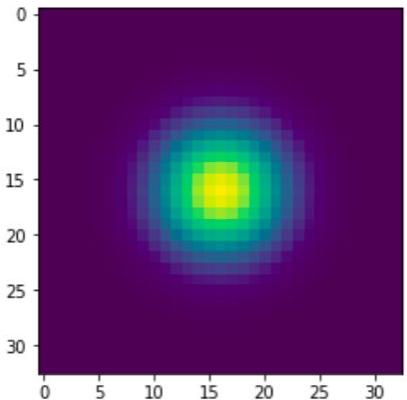


# Размытие



# Gaussian blur

Соседи каждого пикселя усредняются с весами гауссова ядра:



Эффективно реализуется через свёртку.

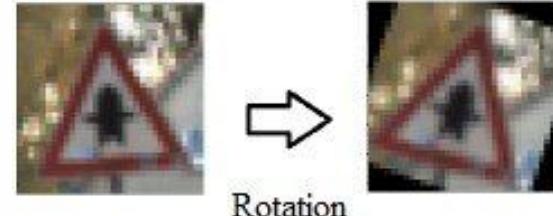


Пример

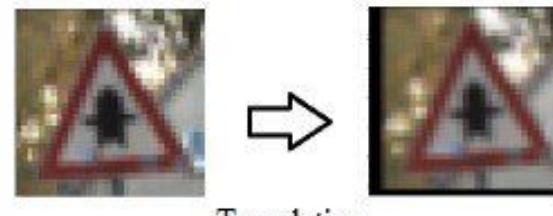
# Геометрические (аффинные)

- Смещение
- Поворот
- Масштабирование
- Обрезание
- Сдвиг
- Отражение

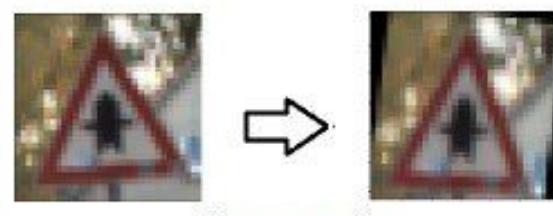
(!) Не забывать изменять метки из корпуса



Rotation



Translation



Shear mapping

# Геометрические аугментации

---

Barrel distortion

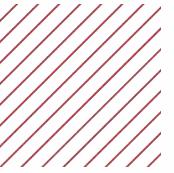


# Геометрические аугментации

---

Elastic augmentations





# Вопросы

---

Спасибо