

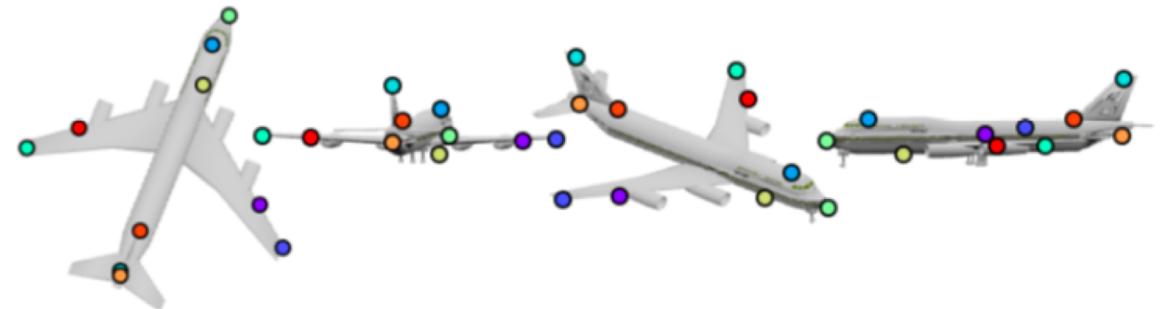
Discovery of Latent 3D Keypoints via End-to-end Geometric Reasoning

Ключевые точки

Ключевая точка – область изображения, которая определяет уникальную особенность объекта

Области применения:

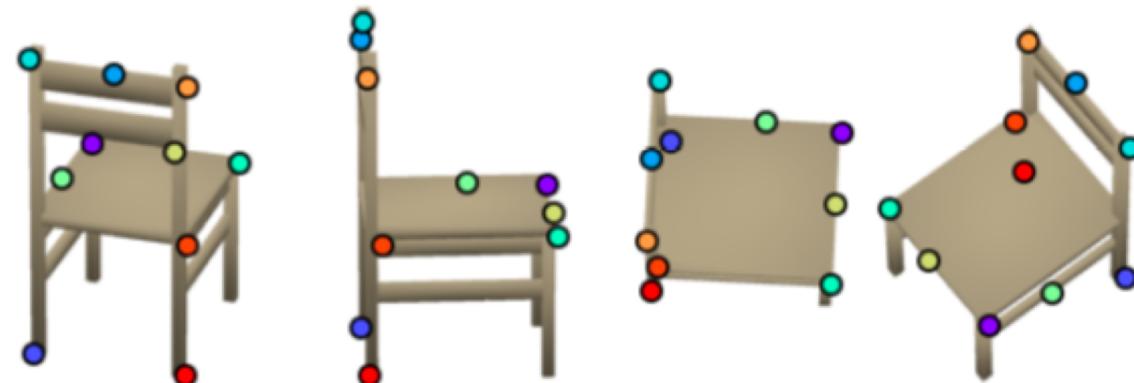
- Сопоставление лиц
- Отслеживание объекта
- Распознавание объекта



Ключевые точки

Преобразования относительно которых
надо получить инвариантность:

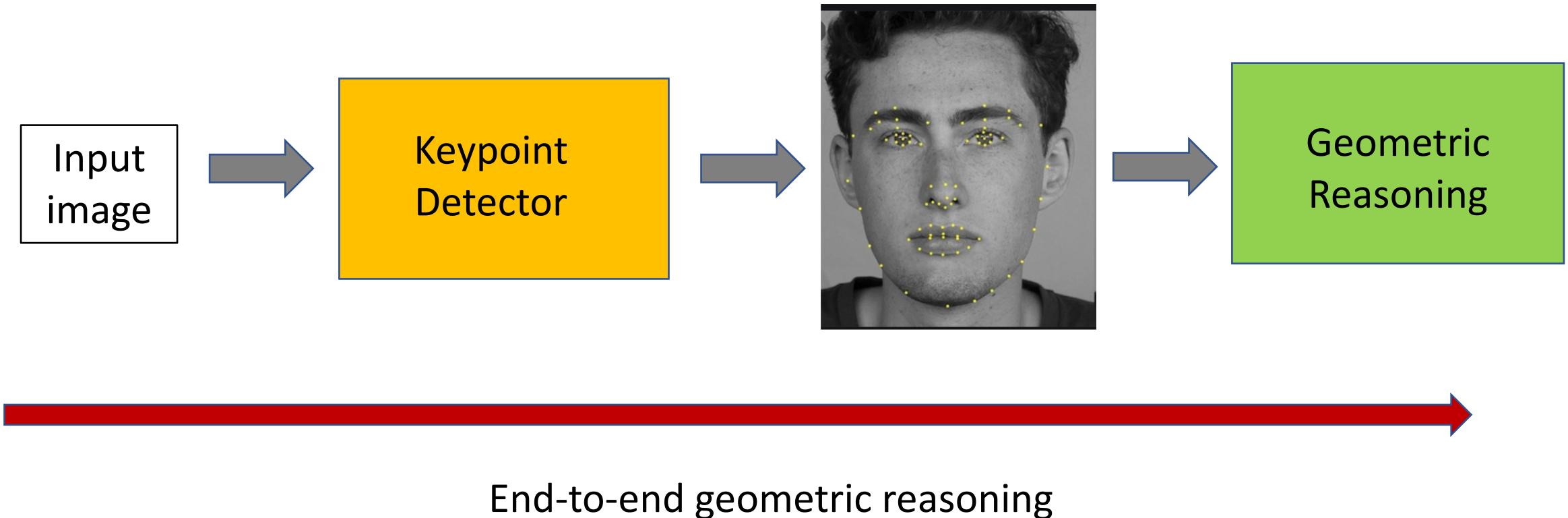
- Изменение яркости
- Изменение размера
(масштабирование)
- Изменения положения камеры (ракурса)
- Вращение
- Смещение



Что выбирать?



Традиционный пайплайн



Обучающие данные

- Две картинки (I, I') одного объекта с разных ракурсов
- Жесткое преобразование T , преобразующее лежащую в основе трехмерную форму из I в I' .

$$T = \begin{bmatrix} R^{3 \times 3} & t^{3 \times 1} \\ 0 & 1 \end{bmatrix}$$

где R – 3D вращение, t - смещение от начала координат

Мы изучаем функцию $f_\theta(I)$, которая отображает 2D-изображение I в список 3D-точек $P = (p_1, \dots, p_N)$ где $p_i \equiv (u_i, v_i, z_i)$ путем оптимизации целевой функции вида $O(f_\theta(I), f_\theta(I'))$.

Согласованность видов (Multi-view consistency)

Цель : ключевые точки отслеживают согласованность разных видов.

Далее везде $[x, y, z]$ – значение координат, $[u, v]$ – координаты пикселя.

Проекция ключевой точки $[u, v, z]$ из изображения I в изображение I' (и наоборот) задается операторами проекции:

$$[\hat{u}, \hat{v}, \hat{z}]^T \sim \pi T \pi^{-1} [u, v, z, 1]^T$$

$$[\hat{u}', \hat{v}', \hat{z}']^T \sim \pi T^{-1} \pi^{-1} [u', v', z']^T$$

Где \hat{u} проекция $u(I)$ на I' изображение, а \hat{u}' проекция $u'(I')$ на первое изображение (I) .

Согласованность видов

Здесь $\pi: \mathbb{R}^4 \rightarrow \mathbb{R}^4$ - операция проекции в координатах камеры. f – фокус камеры

$$\pi([x, y, z, 1]^T) = [\frac{fx}{z}, \frac{fy}{z}, z, 1]^T = [u, v, z, 1]^T$$

Multi-view consistency loss

$$L_{con} = \frac{1}{2N} \sum_{i=1}^N \| [u_i, v_i, u'_i, v'_i]^T - [\hat{u}'_i, \hat{v}'_i, \hat{u}_i, \hat{v}_i]^T \|^2$$

Относительная оценка расположения (Relative pose estimation)

Цель: Восстановить относительного преобразования между данной парой изображений.

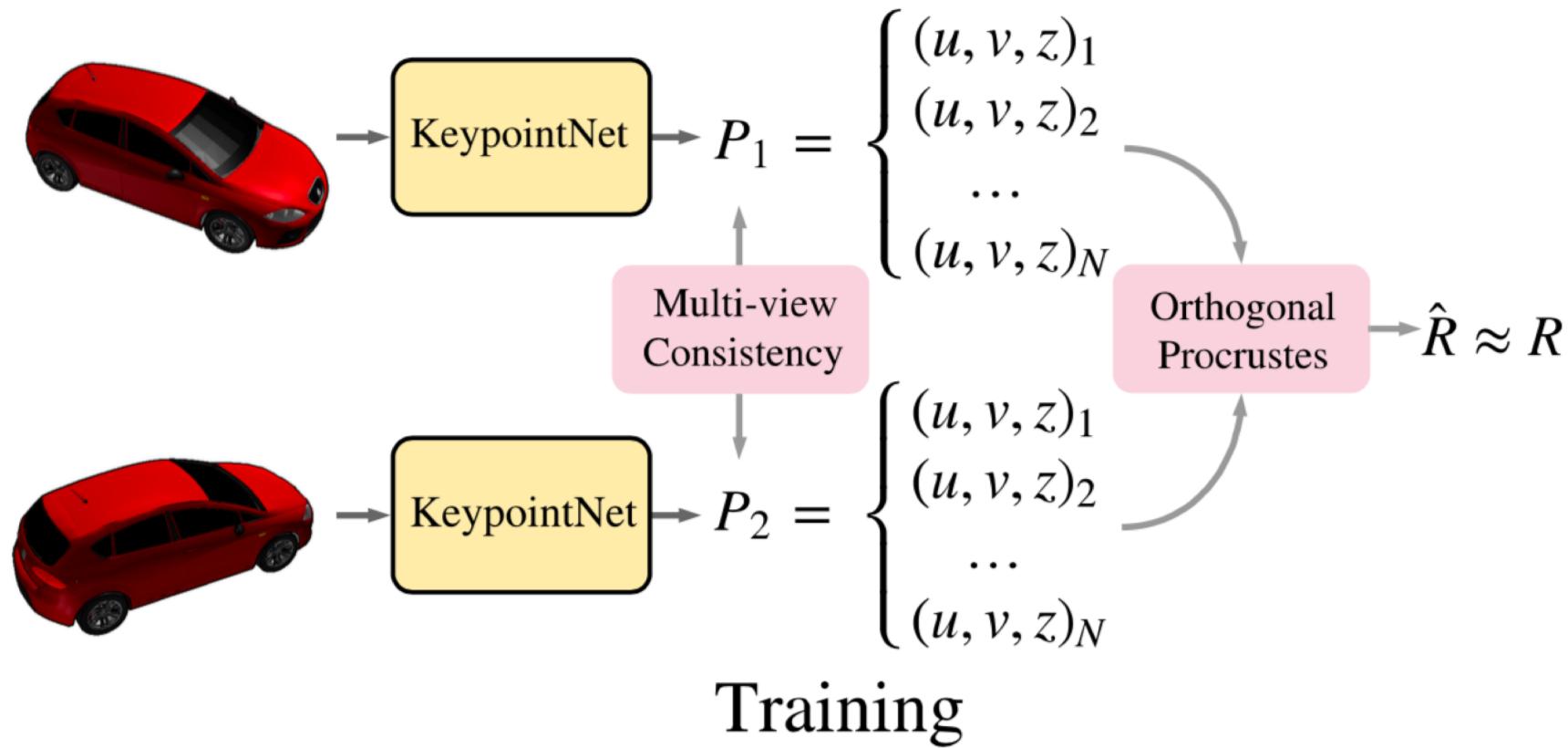
$$L_{pose} = 2\arcsin\left(\frac{1}{2\sqrt{2}} \|\hat{R} - R\|_F\right)$$

Как найти \hat{R} ?

Ортогональная задача Прокруста (orthogonal Procrustes problem).

Решается с помощью SVD разложения

KeypointNet

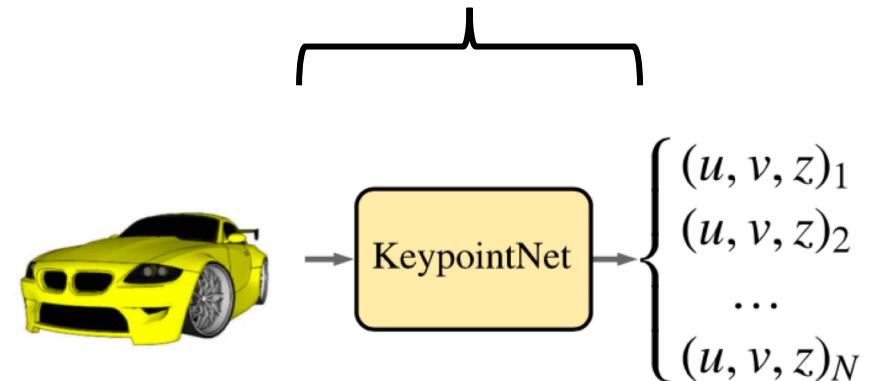


Архитектура KeypointNet

Надо обеспечить **эквивариантность** на уровне пикселей.

Проблема: Обучение CNN без учета этого свойства потребовало бы большого набора данных, который содержит объекты в каждом возможном местоположении

13 layers of dilated convolutions with stride 1



Решение: Сеть выдает карту вероятностей со значением $g_i(u, v)$ – вероятность i -ой ключевой точки оказаться в пикселе u, v .

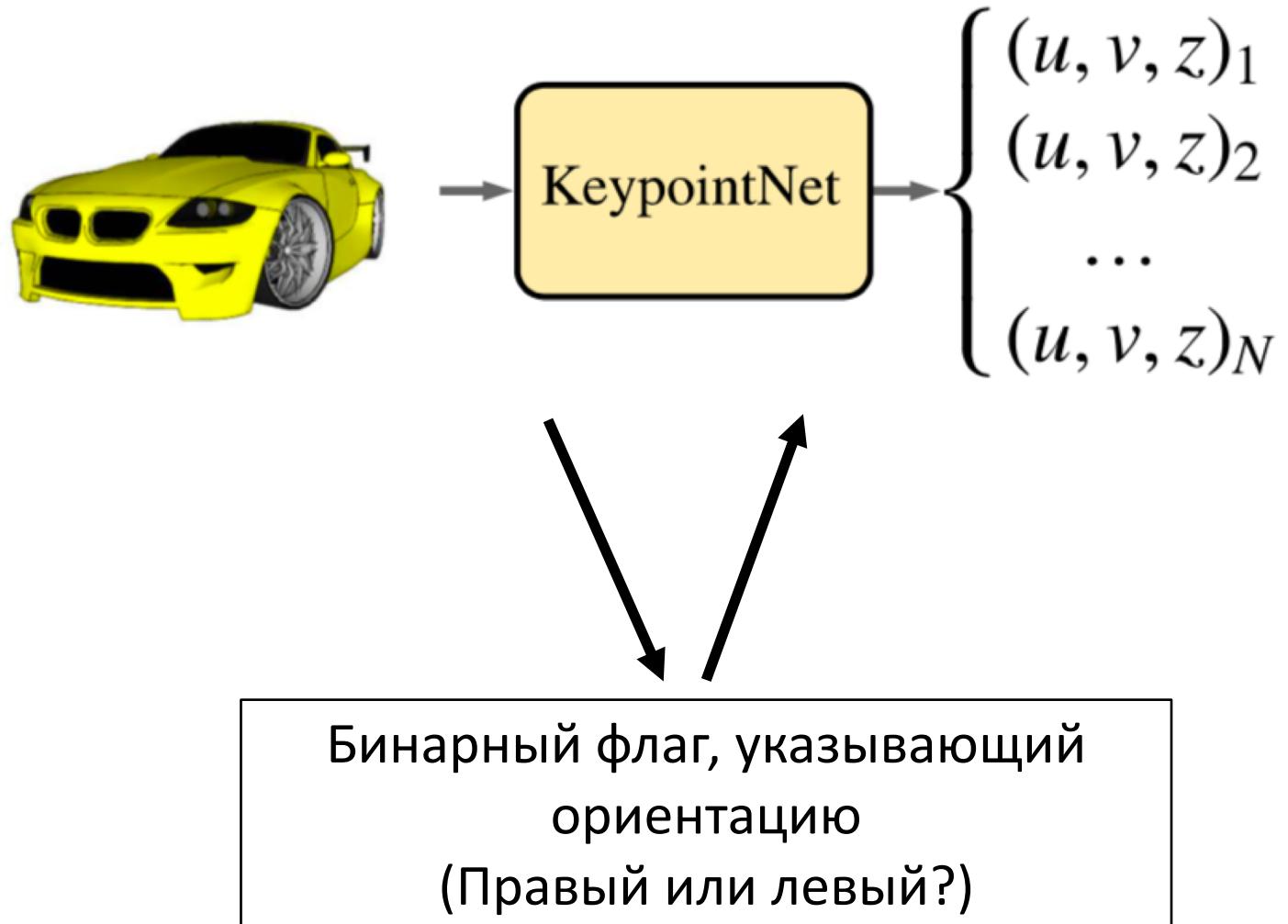
Inference

Чем плохи симметричные объекты?

Проблема: сеть может застрять в локальном минимуме, не имея возможности устранить неоднозначность между двумя симметричными сторонами

Решение: Определять ориентацию объекта дополнительно





Характеристики ключевых точек

- Никакие две ключевые точки не должны иметь одного и того же 3D-местоположения.

Separation loss:

$$L_{sep} = \frac{1}{N^2} \sum_{i=1}^N \sum_{j \neq i}^N \max(0, \delta^2 - \|X_i - X_j\|^2)$$

Характеристики ключевых точек

- Ключевые точки должны находиться в пределах силуэта объекта.

Решение: 1. разрешить только ненулевую вероятность внутри силуэта объекта

$$L_{obj} = \frac{1}{N} \sum_{i=1}^N -\log\left(\sum_{u,v} b(u, v) g_i(u, v)\right)$$

$b(u, v) \in \{0, 1\}$, где 1 означает, что объект переднего плана

2. Поощрять концентрацию пространственного распределения:

$$L_{var} = \frac{1}{N} \sum_{i=1}^N \sum_{u,v} g_i(u, v) \| [u, v]^T - [u_i, v_i]^T \|^2$$

Визуализация

<https://keypointnet.github.io/>



Вопросы

1. Чем подход end-to-end geometric reasoning отличается от традиционных подходов для поиска keypoints? Какие преимущества он дает?
2. Что подается на вход в KeypointNet?
3. С какой проблемой может столкнуться сеть при подаче симметричного объекта на вход при обучении и как авторы статьи решают эту проблему?
4. Какие требования к keypoints должны быть соблюдены и с помощью каких функционалов они учитываются в данной задаче поиска keypoints?

Список источников

- <https://keypointnet.github.io>
- <https://www.youtube.com/watch?v=yxfEoFgqVdk>