hyper motion

DXB

How AI facilitates mobility in Moscow

Dmitry Denisov

Myself

- Work as Data Scientist in Careem (Fraud department)
- Before: Data Scientist in Deloitte
- Interested in Education sphere:

- Conducted a Deep Learning course
https://www.coursera.org/learn/intro-to-deep-learning

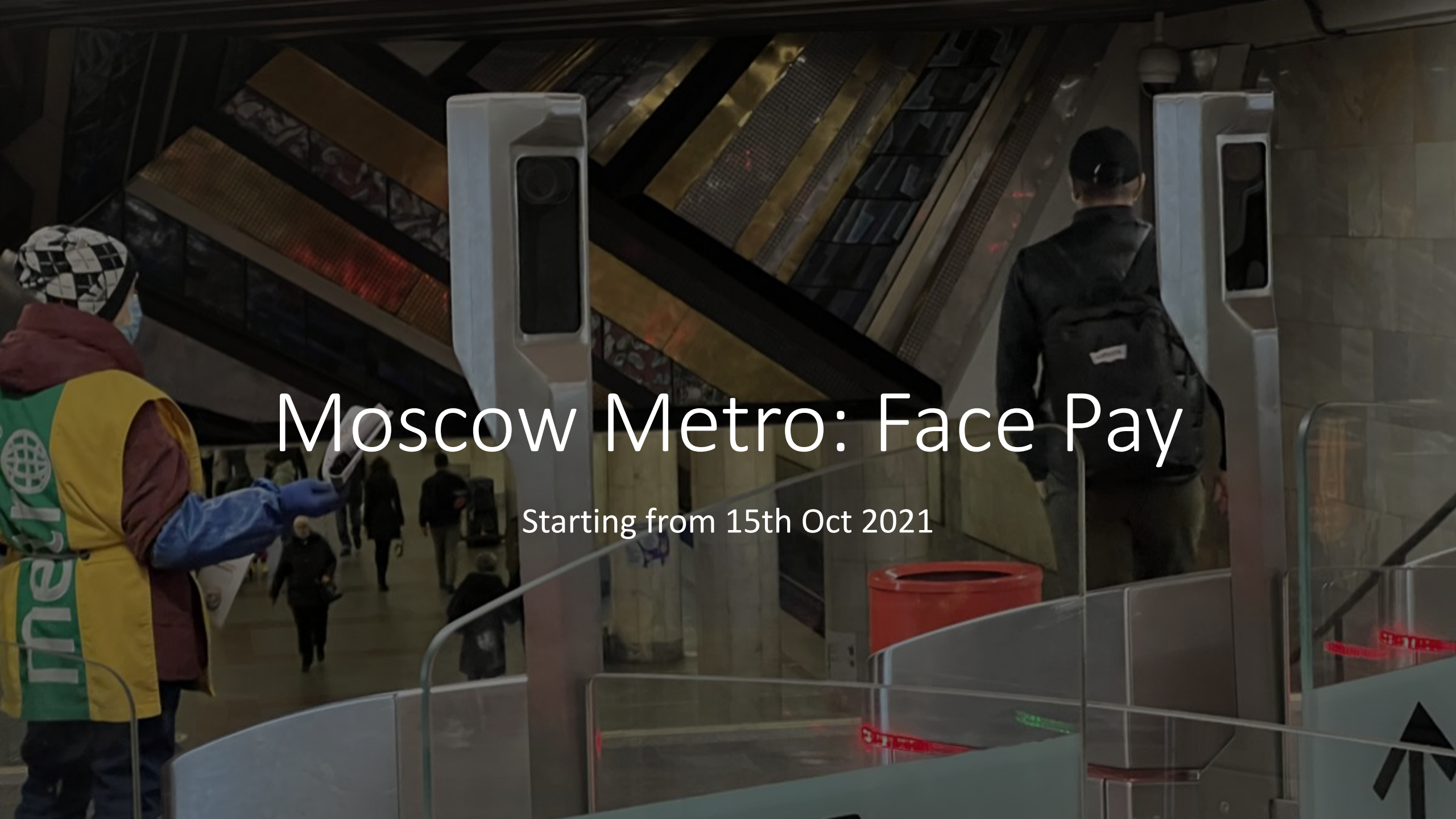
- Course for Open Data Science community:
https://bit.ly/3csuc0g

- Data Science course in Saudi Arabia
- Background: Data Science/Math, Bachelor's in Applied Math and Masters in Data Science

Content

- FacePay in metro
- How FacePay works inside
- Find criminals in metro
- Retail: FacePay
- Bank: ATM confirmation by face
- Transport: different use cases
- Transport: self-driving taxi

Tech!!!



Moscow Metro: Face Pay

Starting from 15th Oct 2021

Main difference with Apple FaceId

Compare input image not with one existining obejct, but with enormous number of objects

Scale:

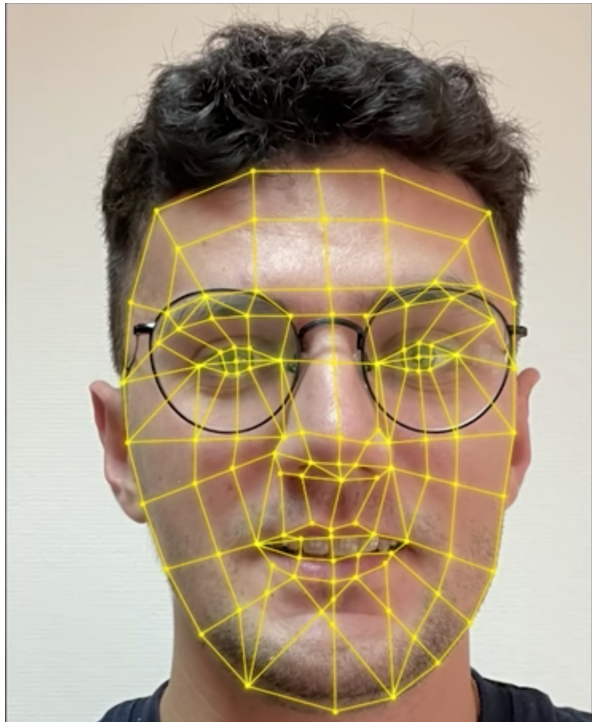
241 stations


Выход
Exit
Выход
Exit
Выход
Exit
Выход
Exit
Выход
Exit

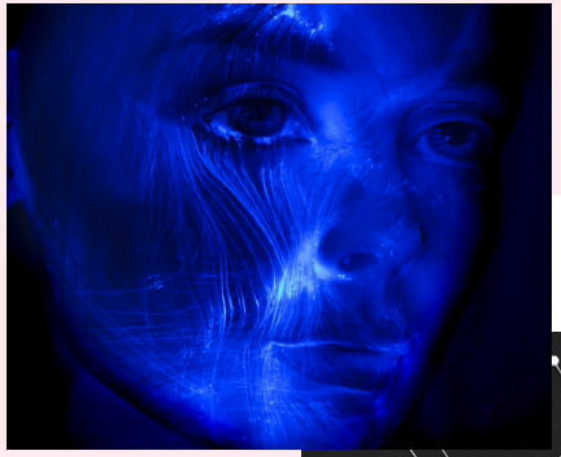

Оплата по биометрии
Face Pay
Оплата по биометрии
При оплат
и карт
Подар
на prive
01 Срок проведен
сентября – 31 ав

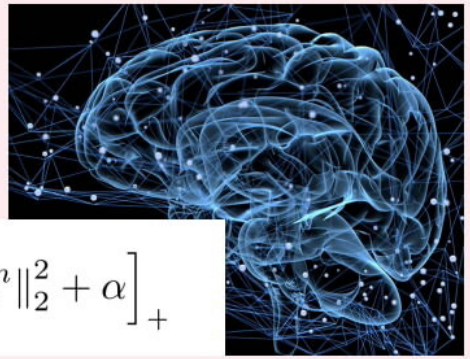
How it works (user experience)

1. Add your photo in Metro Application
2. Come to the turnstile with camera and pass

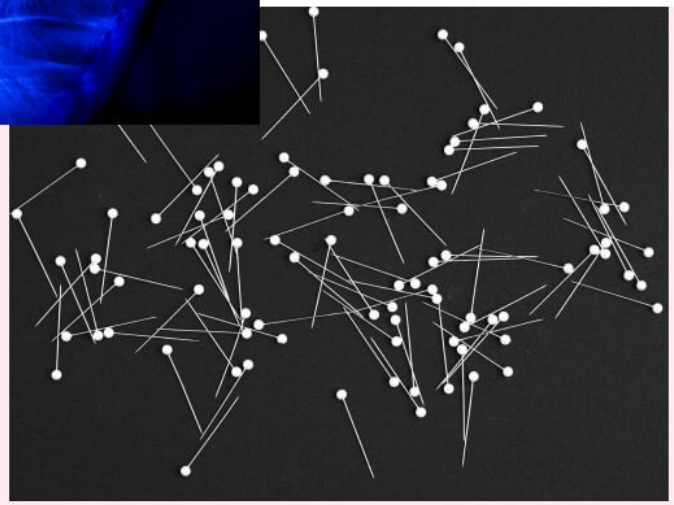


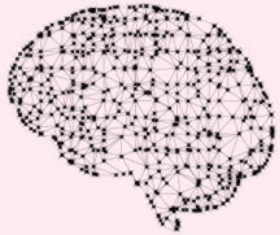
How it works (inside)





$$Loss = \sum_{i=1}^{N} \left[\|f_i^a - f_i^p\|_2^2 - \|f_i^a - f_i^n\|_2^2 + \alpha \right]_+$$



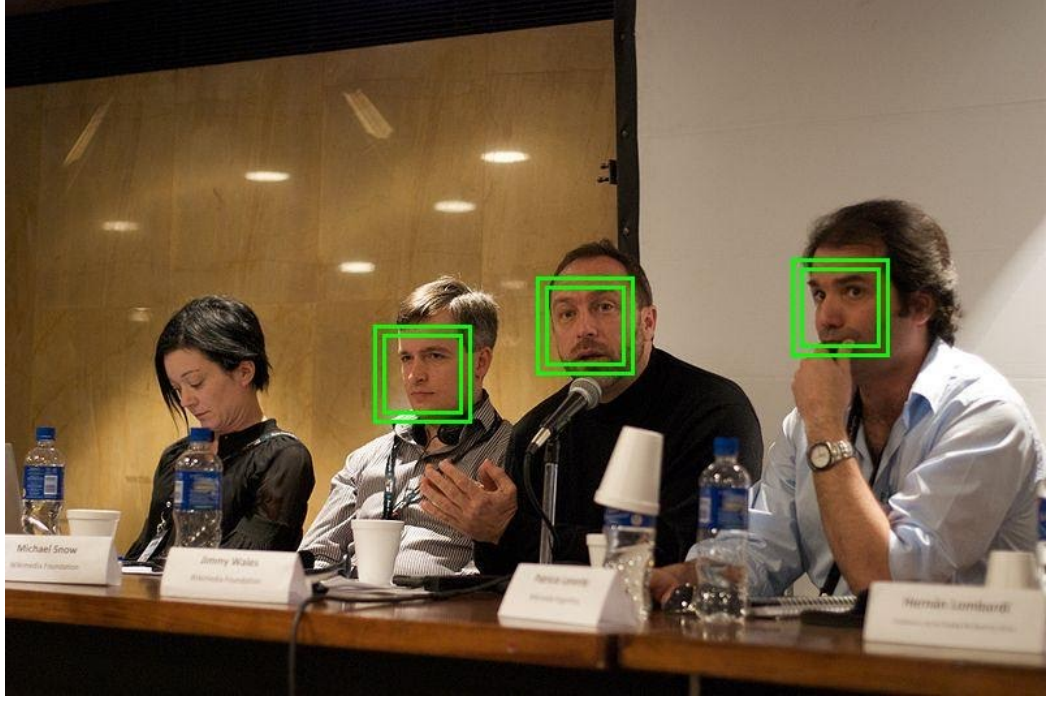


How it works (online)

1. Receive input photo of a customer
2. **Face Detection**: Find customer on the photo
3. **Face Descrition**: On found frame run model, which was offline trained using triplet loss
4. Find in Data Base closest Face Desctiptor

Face Detection

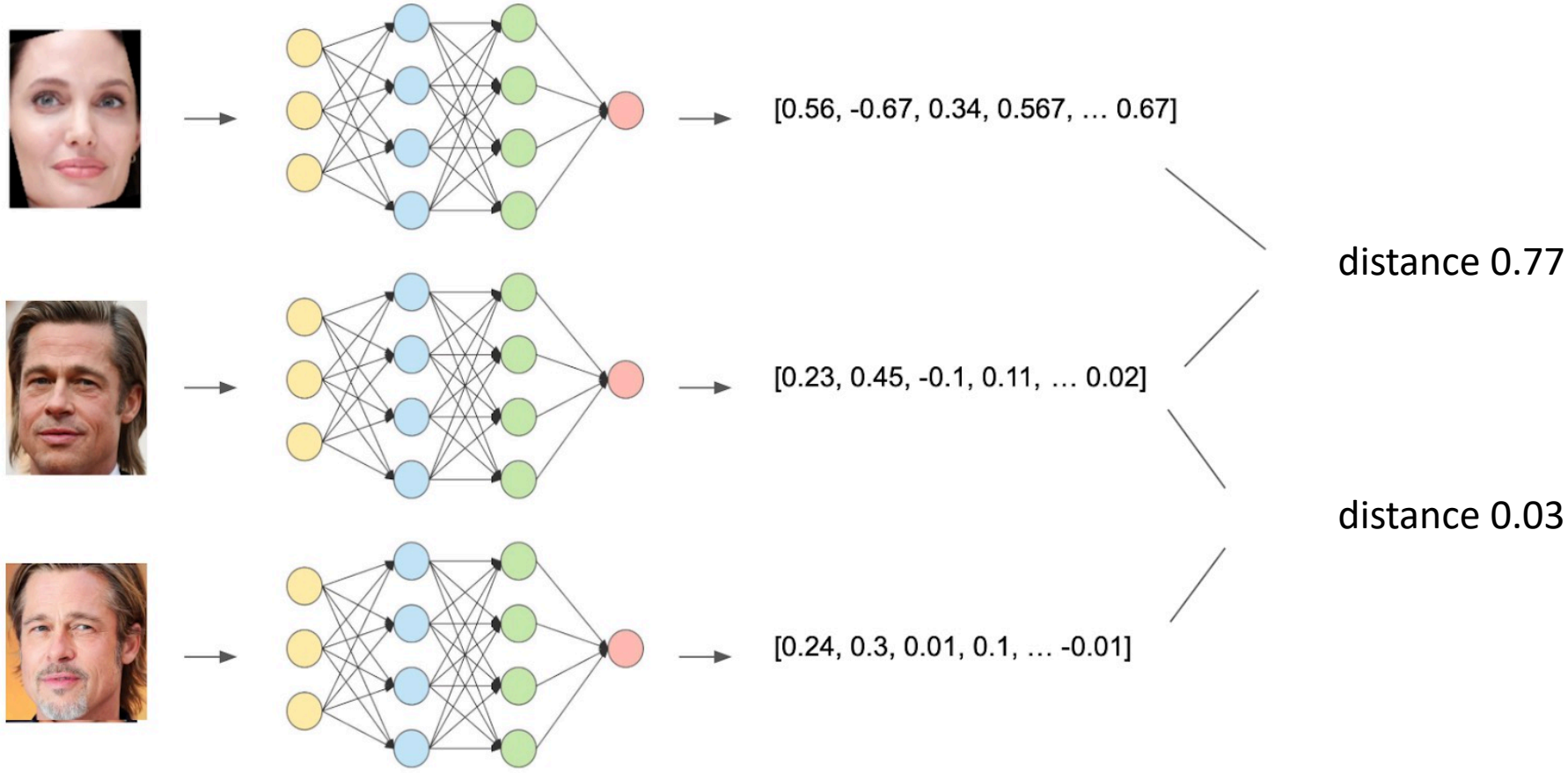
- Step 1
- Step 2
- Step 3



Face Descrition

- Train Neural Network (offline)
- Input – photo of a person
- Output – vector of numbers (called Face Descriptor)
- Face Descriptor property 1: **for different people distance between corresponding vectors should be large**
- Face Descriptor property 2**: approximately the same vector for images of the same face changed depending on hairstyle, makeup, headdress, angle, light and other factors**

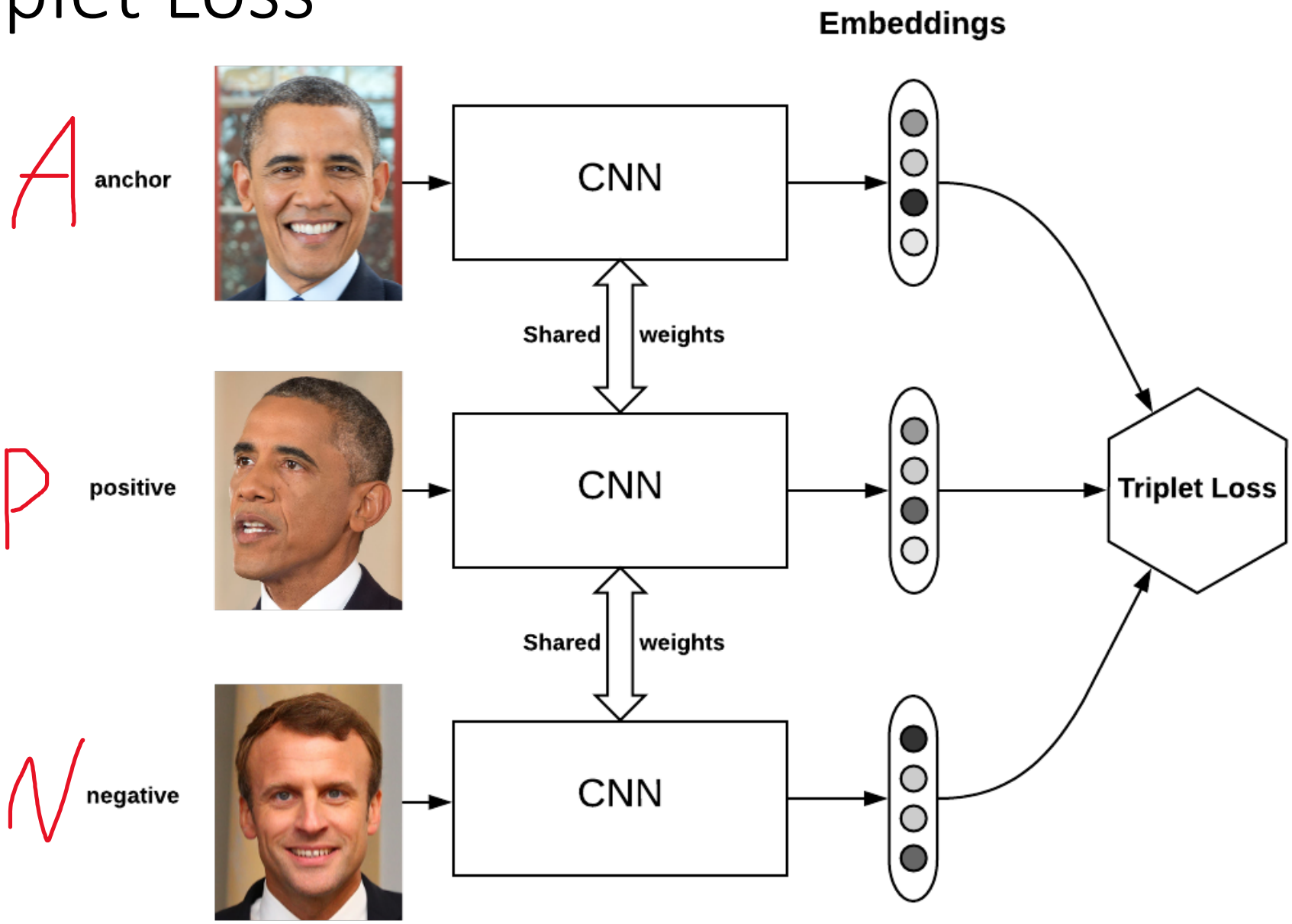
Or visually our idea is:



Idea of training loss:

- **We want: distance(A, P) <= distance(A, N)**
- **To avoid 0 solution: distance(A, P) + c <= distance(A, N), c - constant**

Triplet Loss



How to train such model?

Triplet loss:

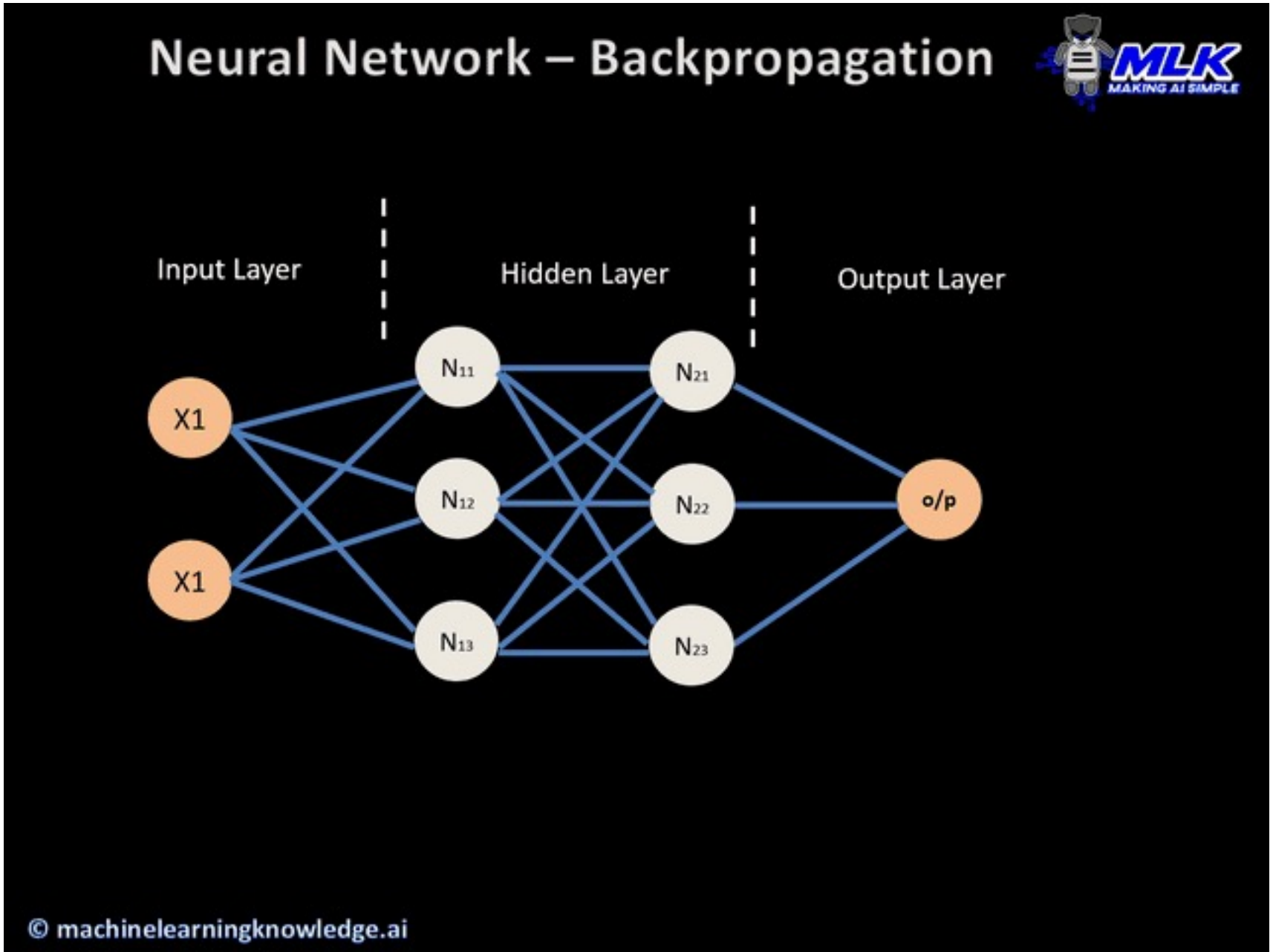
$$\mathcal{L}\left(A, P, N\right) = \max\left(\left\| \mathrm{f}(A) - \mathrm{f}(P)\right\|^2 - \left\| \mathrm{f}(A) - \mathrm{f}(N)\right\|^2 + \alpha, 0\right)$$

Here:

- A = some photo from training dataset
- P = different photo of person A, but taken in different condition (angle, brightness, etc.). Or augmented
- N = photo of another person
- f = Neural Network

Source: https://www.youtube.com/watch?v=Hl8fjhNwOtg

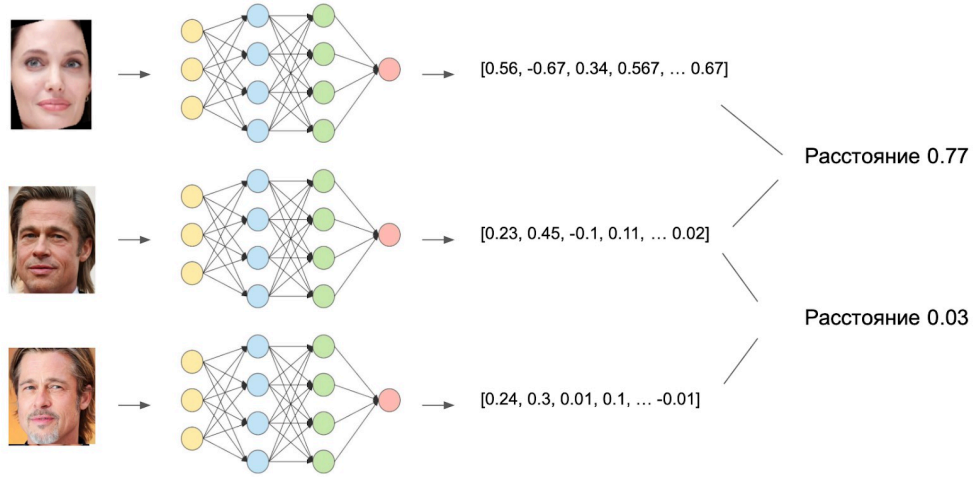
Training schema



How to make training set?

- 10k pictures
- 1k persons
- Generate set of triplets: (A, P, N)
- This is how we get training dataset!

How it works (offline)



1. Train Face Detection model
2. Train Face Description (using triplet loss)
3. Create a Databse, each user will have unique vector, i.e Face Descrition
4. For each new customer we calculte his Face Descrition and add it to the DataBase

Some stats

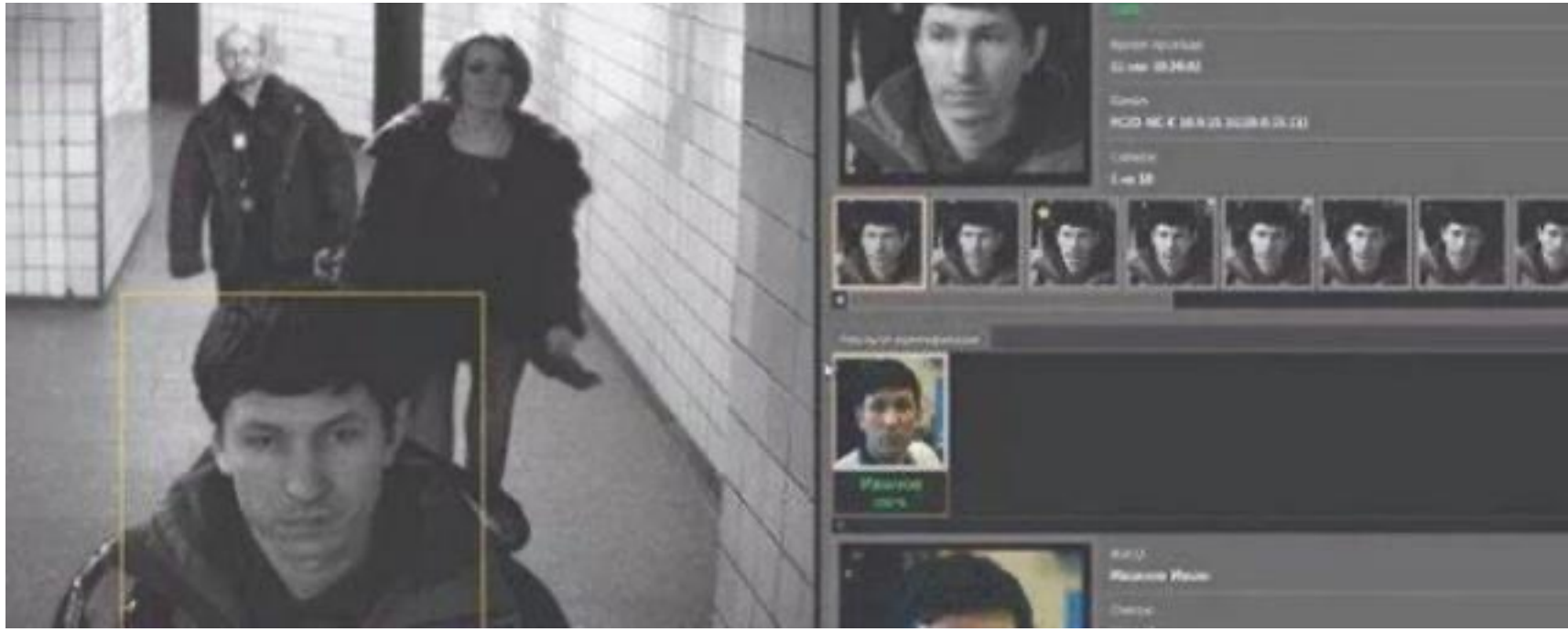
1. 4 months testing period for 85k employees (April – August 2021)
2. 1 month testing for small subset of real customers (August 2021)
3. Available for everyone since 15th Oct 2021
4. Each entrance has one turnstile supports FacePay
5. Accuracy is high (apparently > 95%, but unfortunately exact number is not provided)
6. With probability >70% will be able to identify person in mask

Problems

1. Around 1-2 seconds response => problems in rush hour
2. Makeup
3. Different seasons (summer/winter)
4. Fraud: someone can remove his head in the correct moment such that you will pay

Find criminals + lost people in metro (Project Sphere)

- Working starting from Sept 2020
- For 2021: 1519 criminals arrested
- For 2021: 221 lost people were found

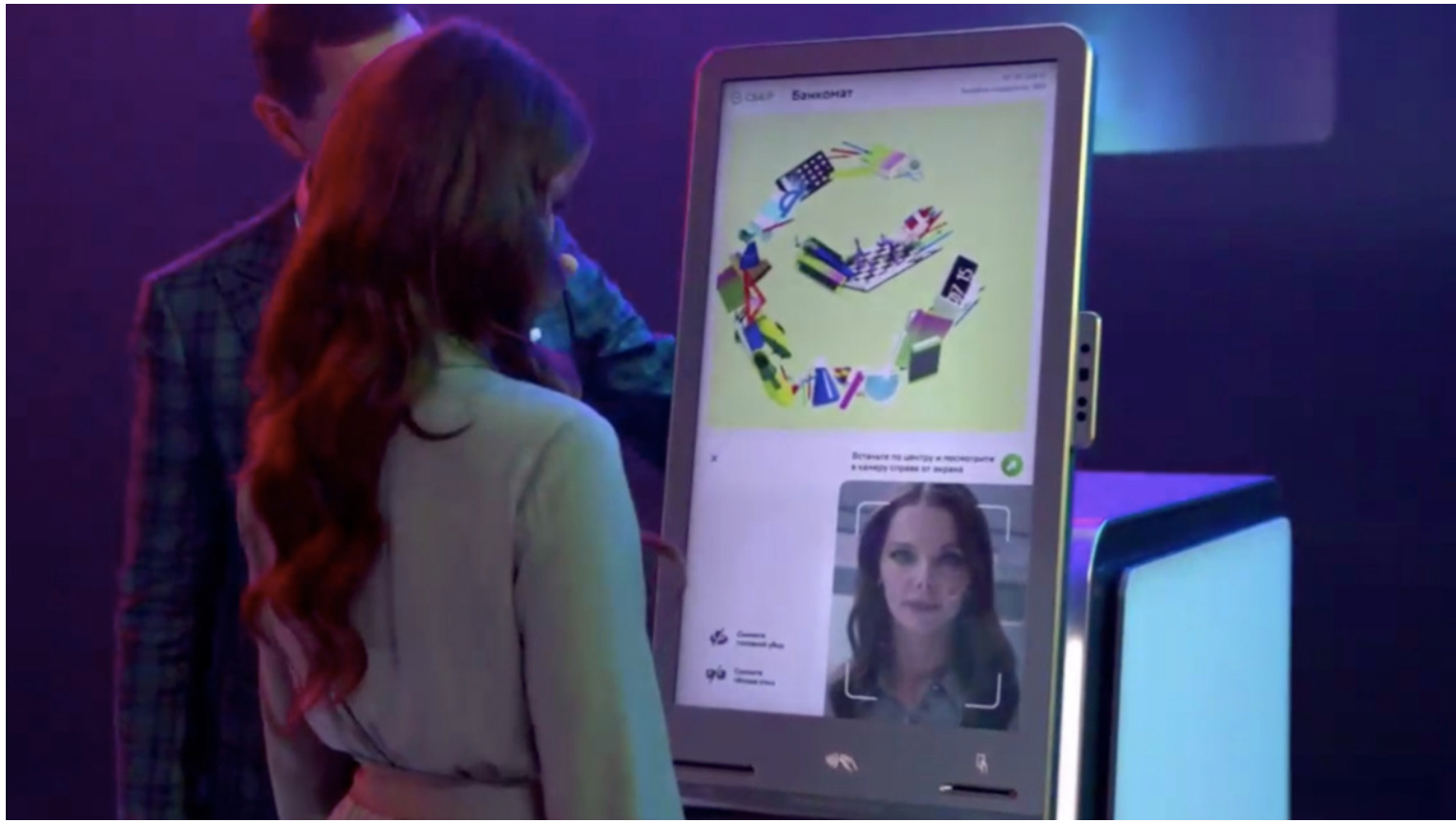


Retail: Perekrestok

- Launched in test mode in 200 supermarkets
- Launched in March 2021

Bank: Sber

- Whenever you call contact center: Voice confirmation
- ATM: face id confirmation
- On July 2021 42% ATM supports identifying by face or 10% of whole ATM market



Transport

- Plate numbers + automatic fines: works from Dec 2018
- Fine for not using seat belt: July 2021: 62 cameras support it

Transport: parking

- Working from 2015

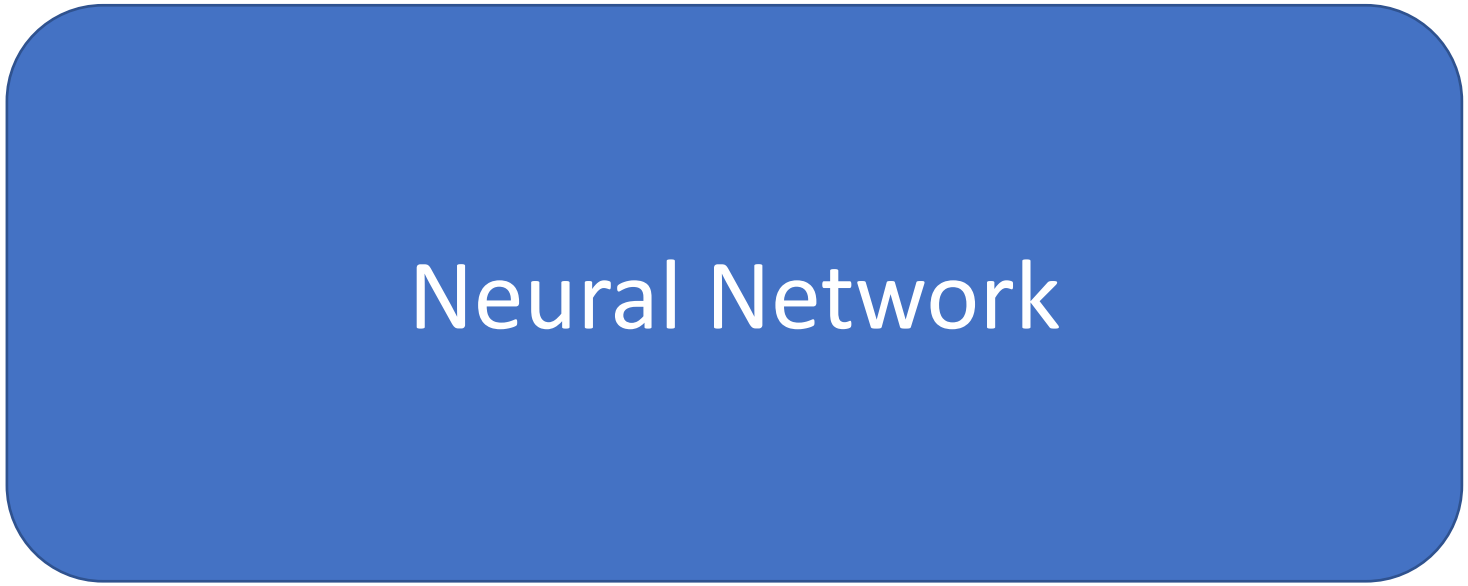
Transport

Identify turned off headlights: 17th October 2021: 1st camera enabled

Transport: Self-driving cars

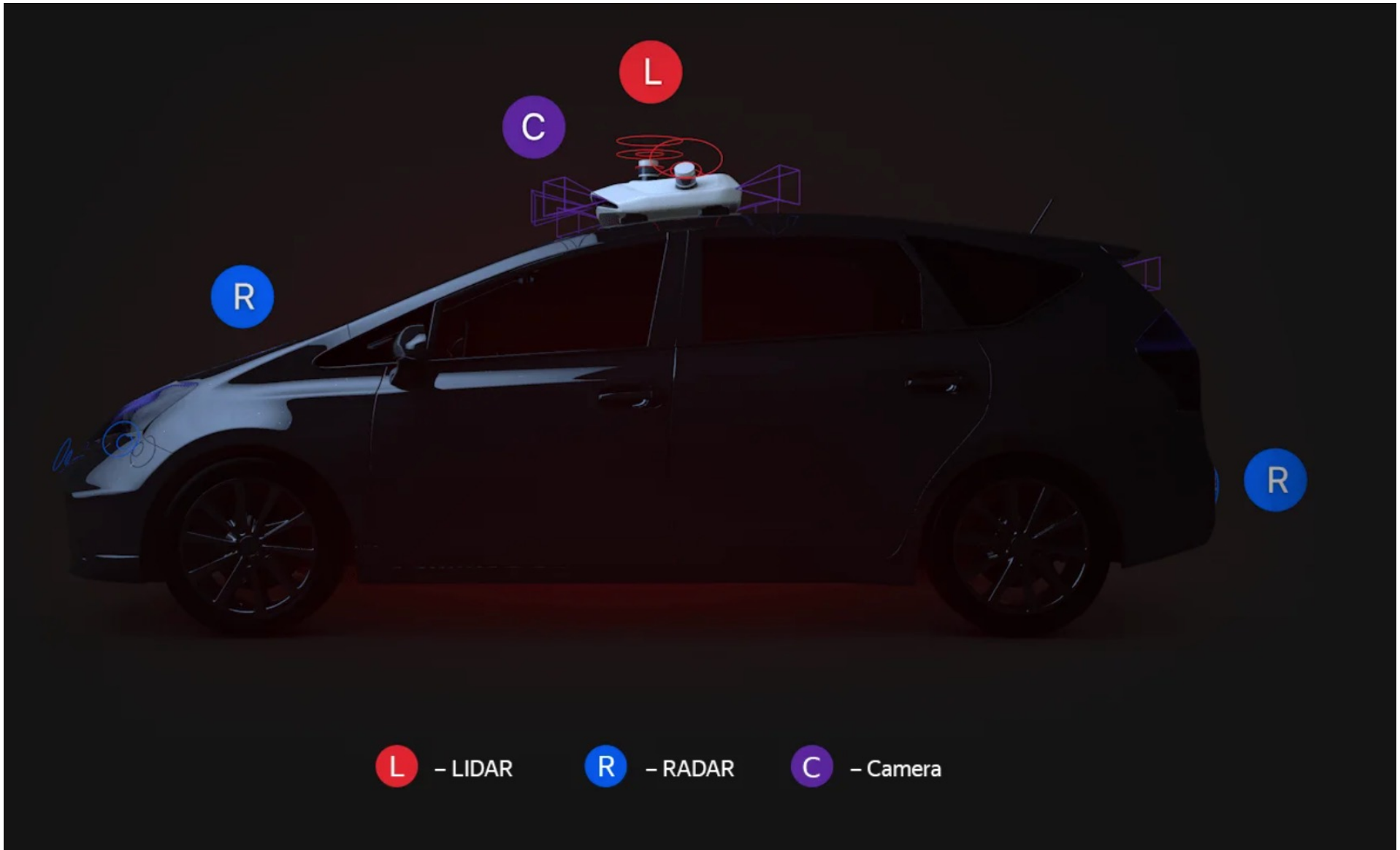
Self driving car inside

Self-driving cars: end-to-end approach



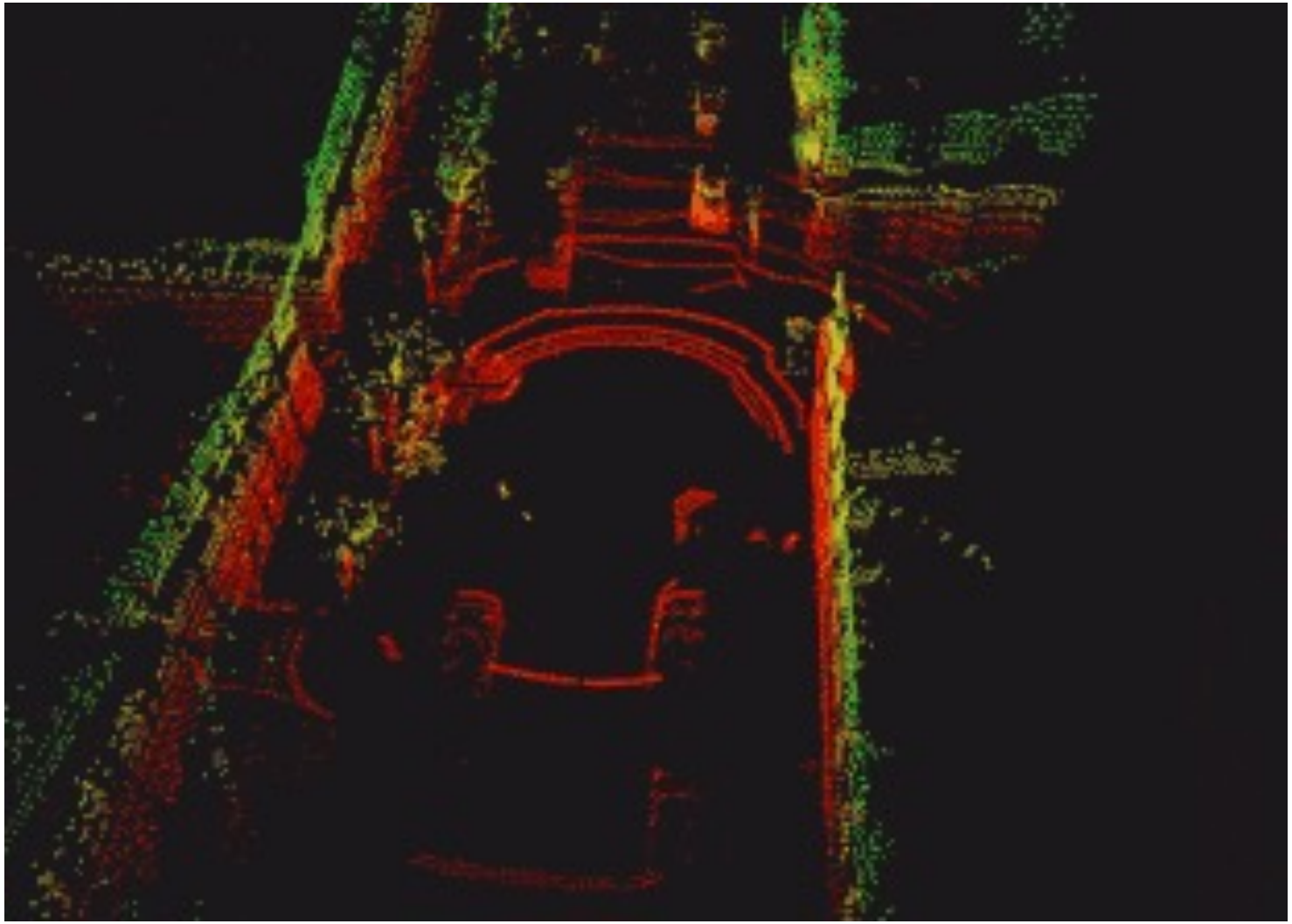
Classic approach: Lidars + cameras + radars

All sensors:



Lidar

- Sends laser beams, receives it back
- Based on time can calculate distance to the point



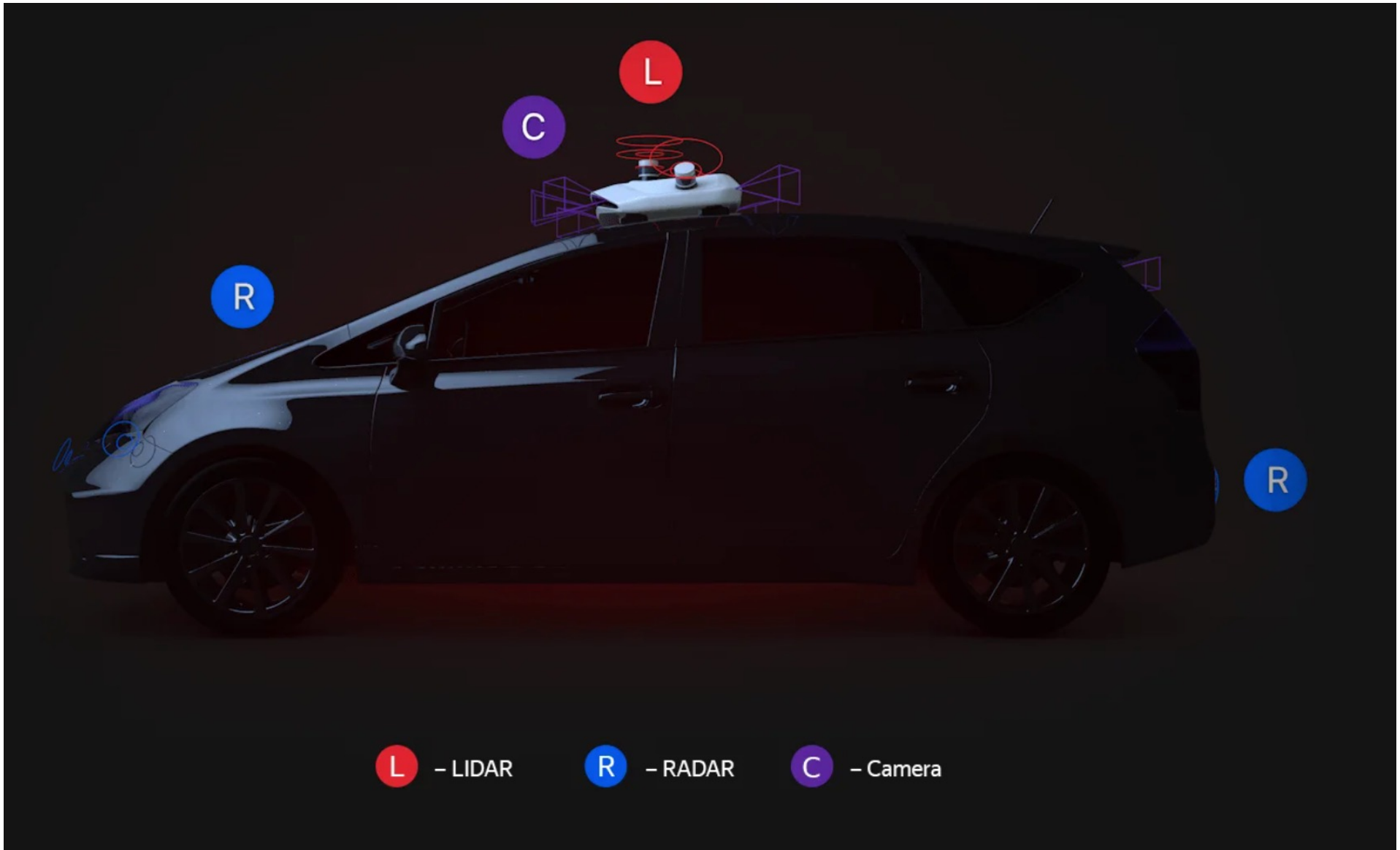
Lidar: advantages

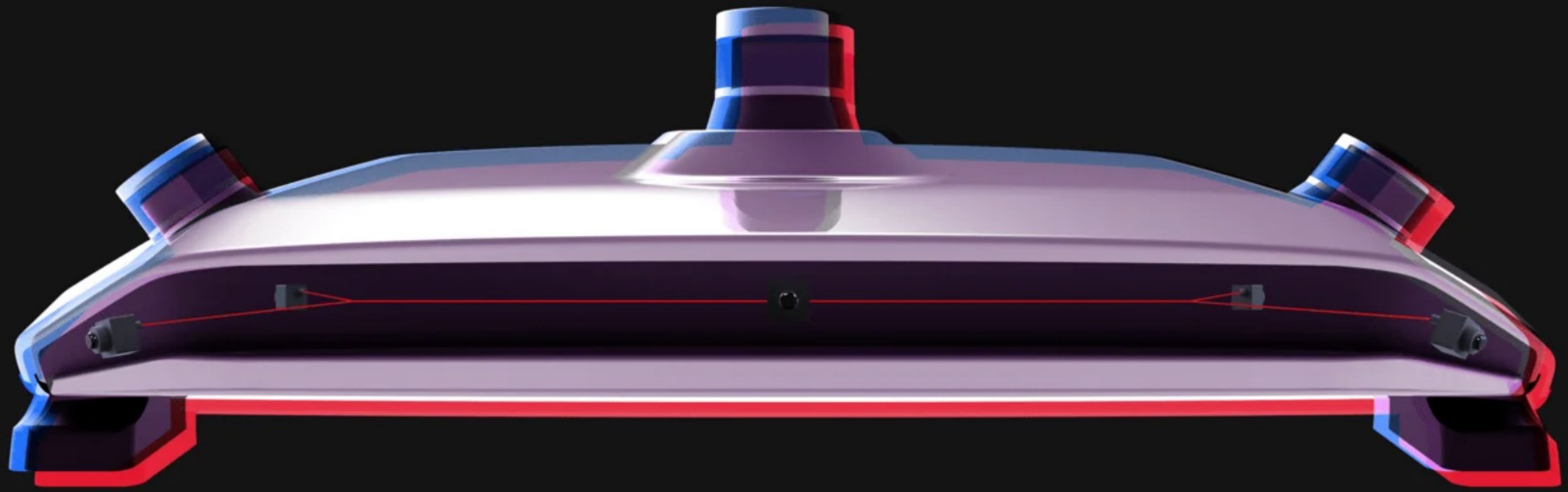
- Very accurate
- High density of points
- Works out of the box

Lidar: why can’t we just use it?

- rough road
- traffic fumes
- Complete black car won’t be visible, because rays will be absorbed
- We don’t separate objects by classes
- Reflection from puddles
- Weather: rain, snow
- Price!!! >20k$

Camera:



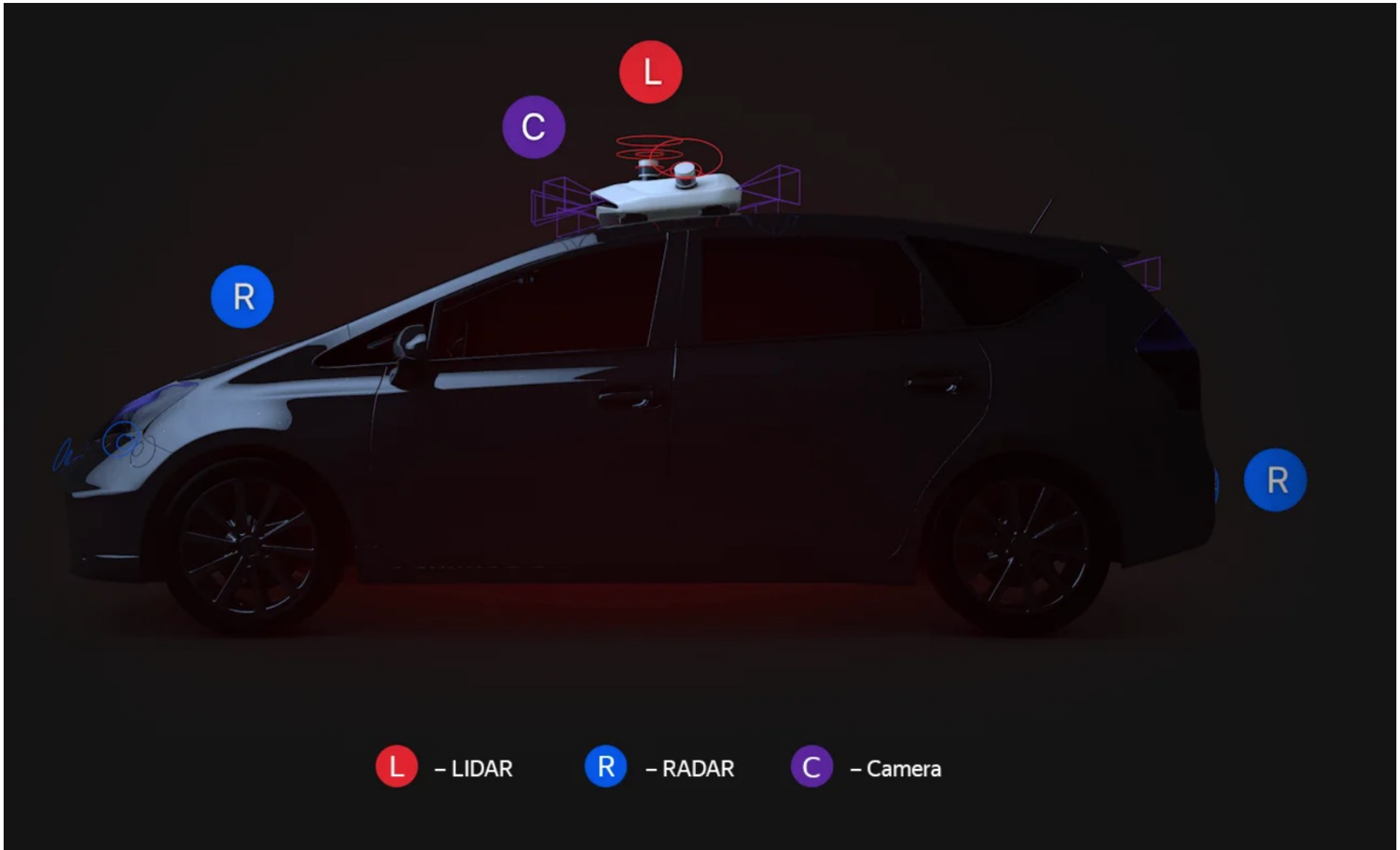


Cameras

Cameras point in every direction from the car, acquiring visual information about traffic signals, road surface markings, and beyond. This helps the car form a better overall picture of its surroundings than a human ever could.

Camera: semantic segmentation

Radar:



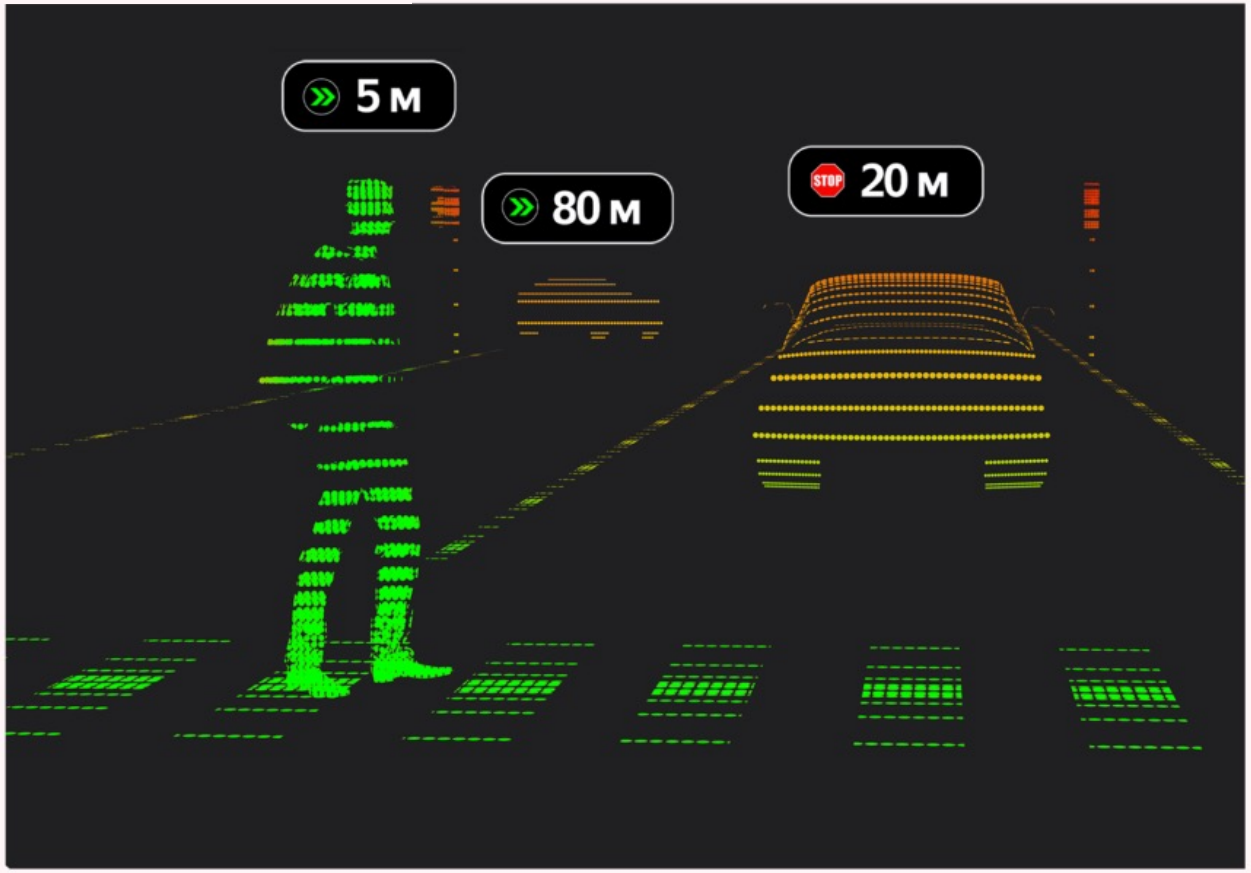
Radar

RADAR works just like LiDAR, with the big difference that it uses radio waves instead of laser or light. It transmits radio waves from a rotating or stationary antenna and measures the flight time of the reflected signal.

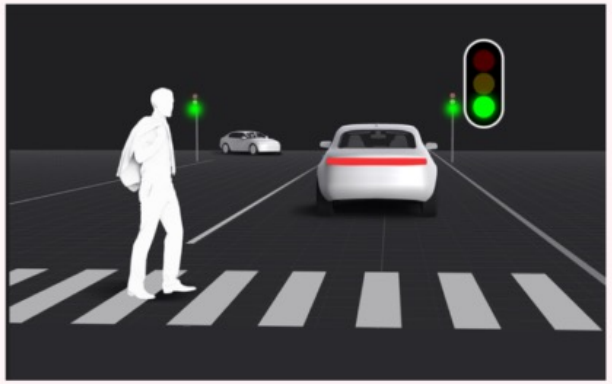
Radar: advantages

- Speed of objects
- Diretions of objects
- Work out of the box

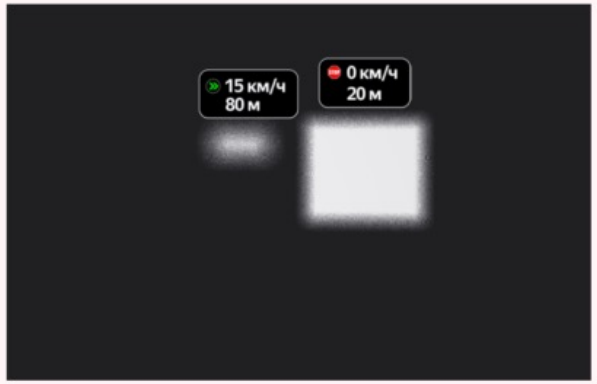
Lidar image



Camera image



Radar Image



Full scope

Localization	Detection	Planning
➢GNSS ➢Odometry ➢ICP	➢2D/3D objeccts ➢Semantic Segmentation	➢Trajectory ➢Traffic Rules

Full scope

Localization

- GNSS
- Odometry
- ICP

Detection

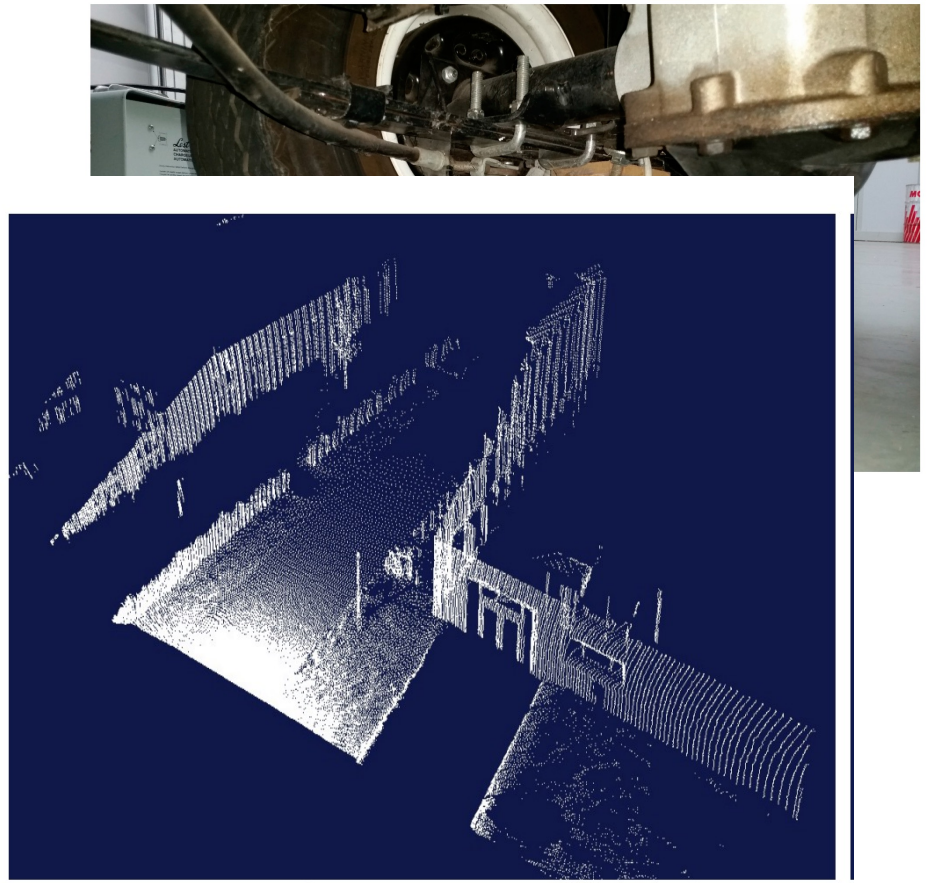
- 2D/3D objeccts
- Semantic Segmentation

Planning

- Trajectory
- Traffic Rules

Localization

1. GPS: 5-10meters, in city 20m -> GNSS RTK (Real Time)
2. Odometry
3. ICP: iterative closest point

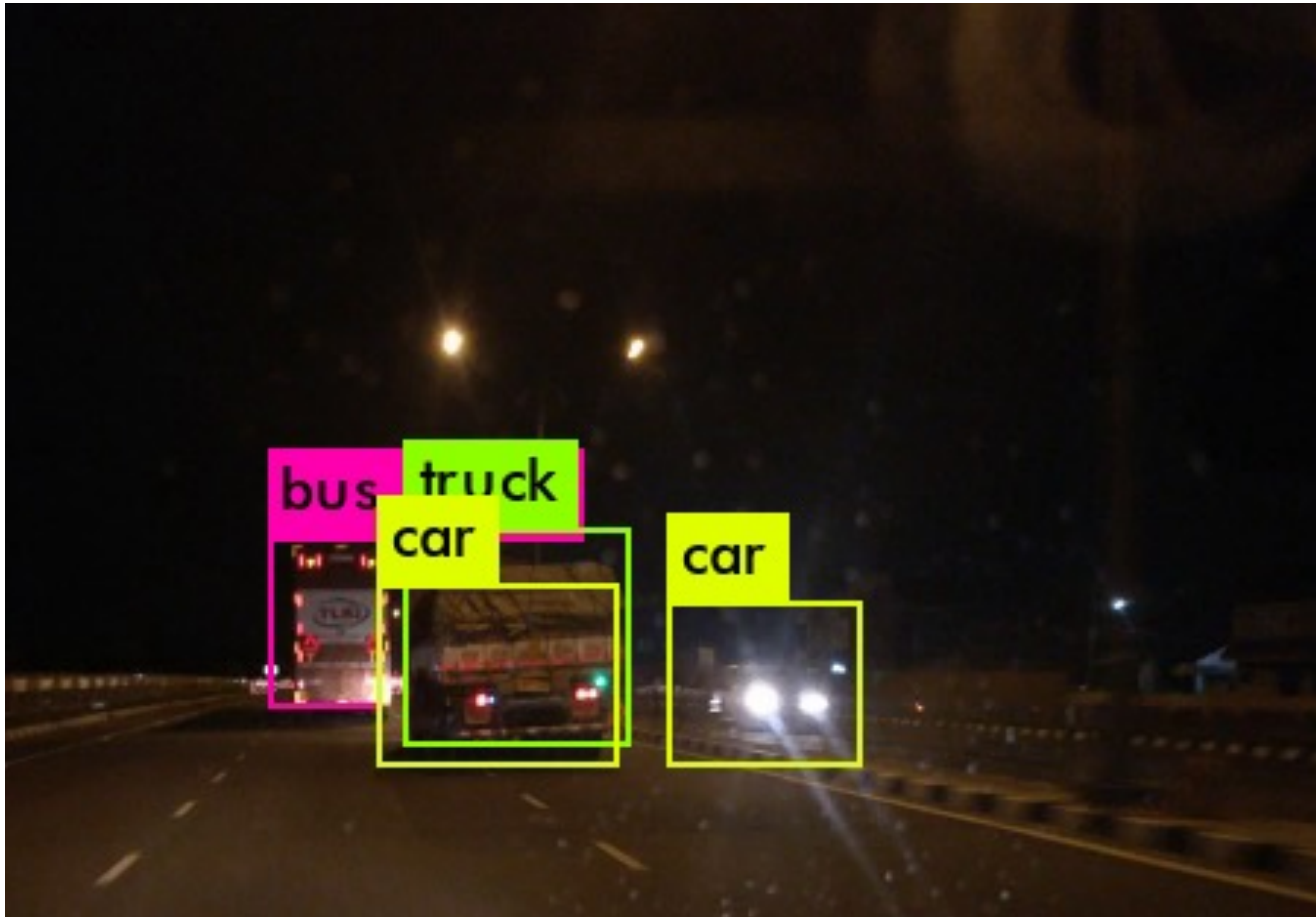


3D Scan (~200.000 Points)

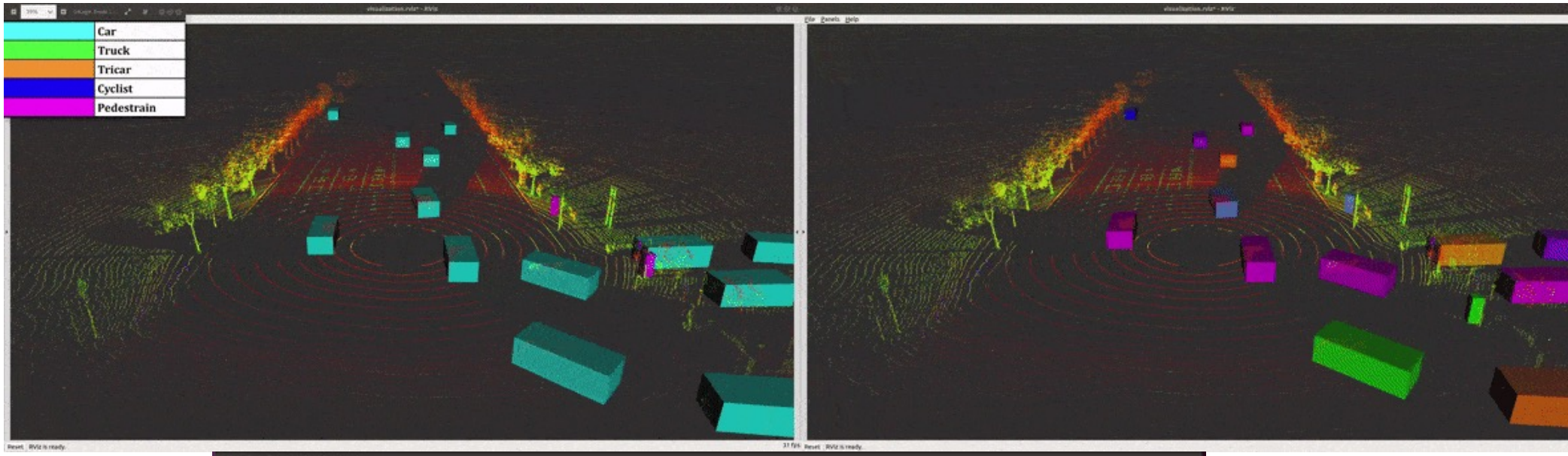
Full scope

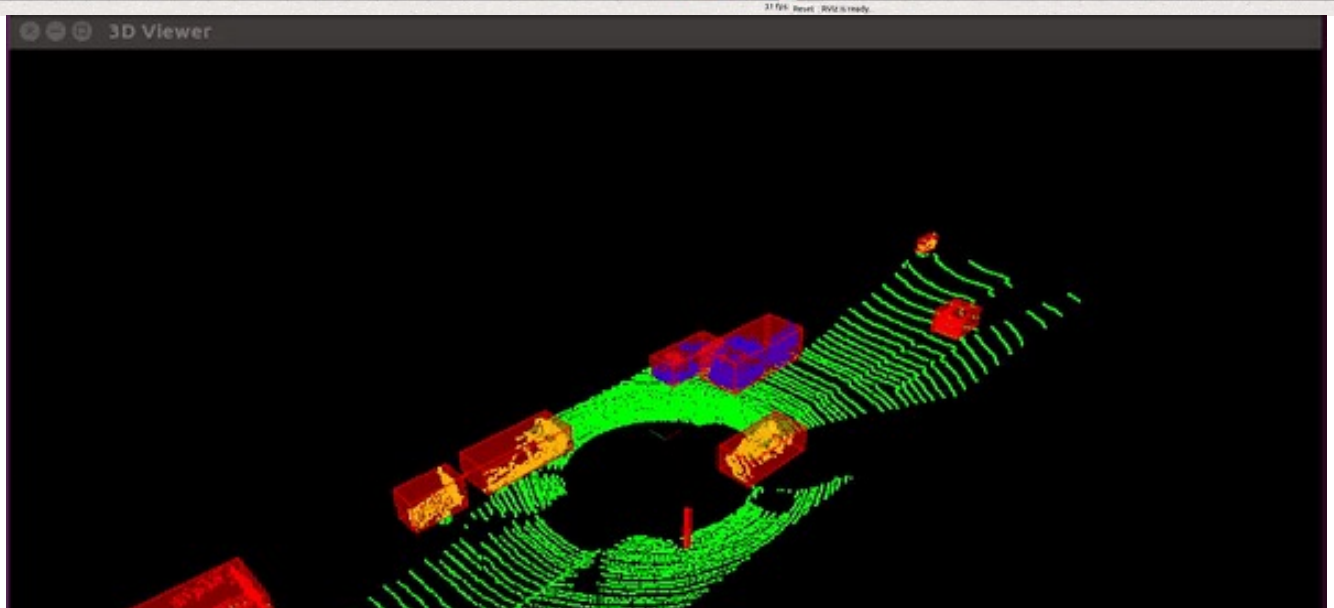
Localization	Detection	Planning
➢GNSS ➢Odometry ➢ICP	➢2D/3D objeccts ➢Semantic Segmentation	➢Trajectory ➢Traffic Rules

Detection: 2D objects

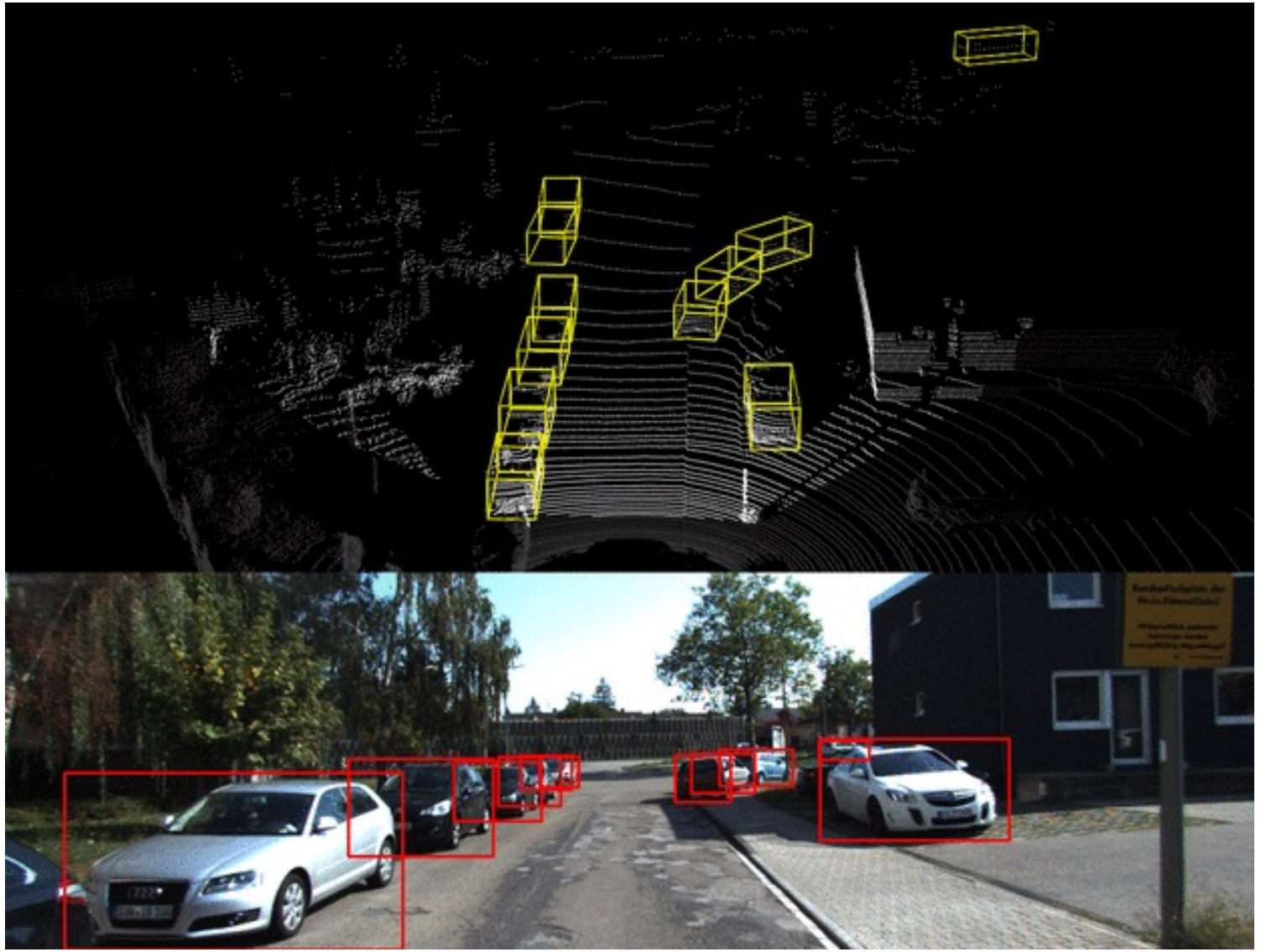


Detection: 3D objects





Detection: 2D and 3D together



Full scope

Localization	Detection	Planning
➢GNSS ➢Odometry ➢ICP	➢2D/3D objeccts ➢Semantic Segmentation	➢Trajectory ➢Traffic Rules

Planning

Traffic analysis

Take decision

Build trajectory

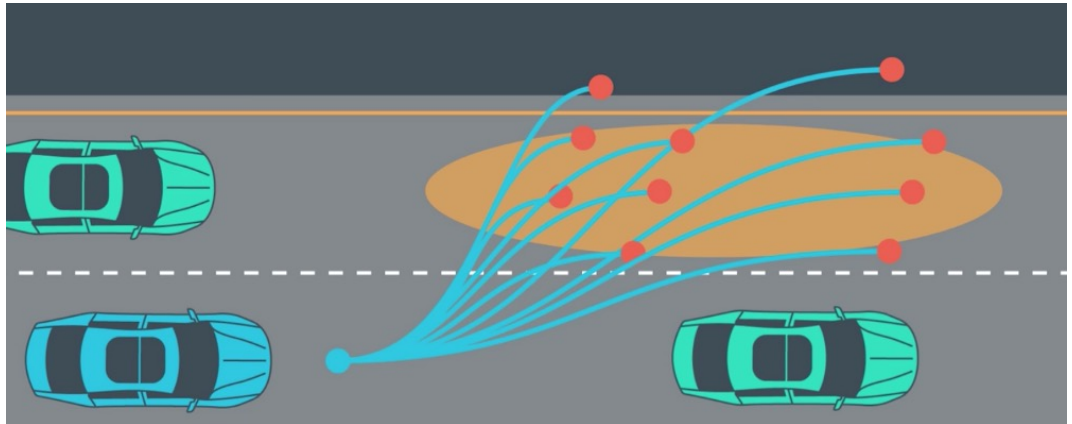
Complete trajectory

Should we give way for someone? Does traffic light allow to move? What is this pedestrian going to do?

Change the line/wait until pedestrian will cross the road/slow down/go into parking mode/…

Build possible trajectories and choose the best one

control



Self-driving Taxi:

- 2017: start of developing
- 2018: first inter-city trip in total 780km
- 2018: test mode for taxi in one of Russian cities
- 2018-2021 test mode on couple of regions in Russia (not taxi, just self driving)
- Autumn 2021: for chosen customers taxi service will be available, around 170 cars in total

Self driving cars: more …

Thank you! Questions?

GitHub: https://github.com/DmitriiDenisov

LindedIn: https://www.linkedin.com/in/dmitry-denisov-022102103/

Web: https://ddenisov.com/