

Mumbai: Accident victim dies as ambulance gets stuck in traffic jam

| Thursday | 21st April, 2016

Choking life: 12 on way to hospitals die in traffic jams

TNN / Updated: Oct 18, 2016, 09:49 IST

Lives at risk from long ambulance waits, say paramedics

By Nick Triggle
Health correspondent

| 11 November 2021 |  Comments

New victims of Bengaluru's traffic jams: Patients in ambulances

AUTOMEDIC

PROJECT PHOENIX

NHS patients dying in back of ambulances stuck outside A&E, report says

Ambulance Stuck in traffic, Scary Right?

Infant dies as ambulance gets stuck in traffic amid protest by contractual employees

City plights: Stuck between life and death on Delhi's roads

PROTECTING LIVES

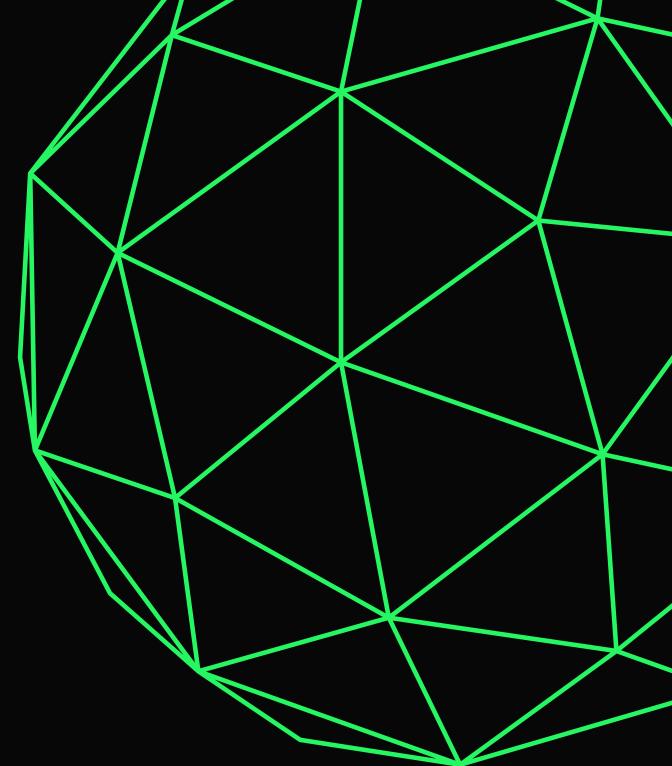
WITH SMART VISION

A look at how technology can help save lives!

Discussion points

Key topics covered
in this presentation

- The problem
- The solution
- Technologies used
- USP & Features
- Market Validity & Target Market
- Prototype Implementation
- Business model



THE PROBLEMS

Ambulances getting stuck in a traffic congestion while carrying a patient.

Cars willing to get out of the way but unable to due to timed traffic lights.

The present solutions being quite expensive to carry on a large scale.

THE SOLUTION

AUTOMEDIC is a python autonomous system capable of:

STEP 1

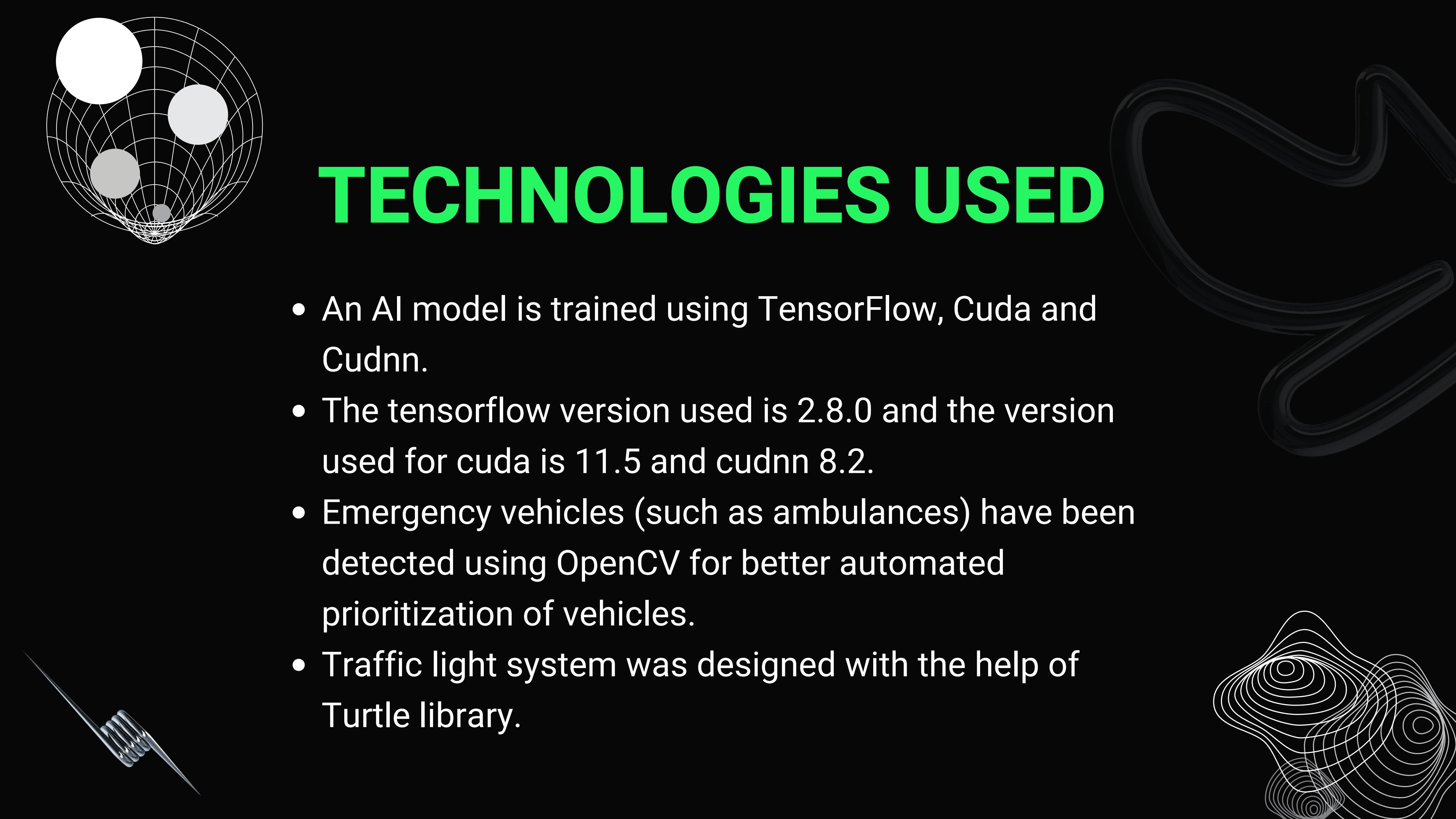
Detecting emergency vehicles from an appropriate distance (1.0 km)

STEP 2

Turning the traffic light from red to green (if the light isn't already green)

STEP 3

Allowing the ambulance to pass safely!



TECHNOLOGIES USED

- An AI model is trained using TensorFlow, Cuda and Cudnn.
- The tensorflow version used is 2.8.0 and the version used for cuda is 11.5 and cudnn 8.2.
- Emergency vehicles (such as ambulances) have been detected using OpenCV for better automated prioritization of vehicles.
- Traffic light system was designed with the help of Turtle library.



FEATURES



Detect vehicles from a km ensuring enough time for the lights to change safely.

No upfront cost.

95% accuracy.

Enhance the pre-existing CCTVs cameras.

MARKET VALIDITY & TARGET MARKET

Can be used in collaboration with the government.

Can be used to provide multi-city mapping.

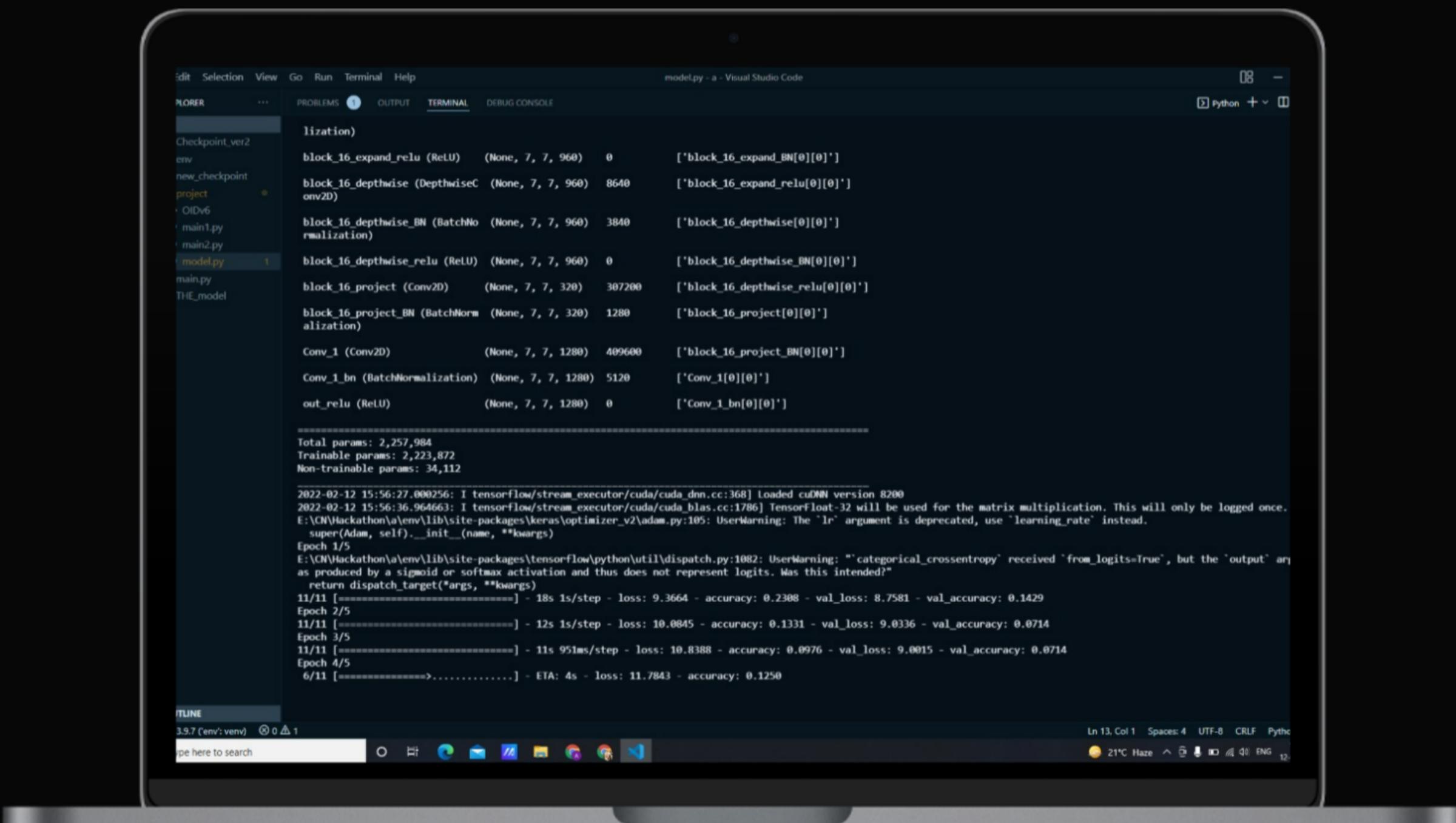
Can be used by private healthcare enterprises.

Further development of the technology can also provide info such as identification of traffic accidents.

The technology can even be used in traffic management and detection of other vehicles.



PROTOTYPE IMPLEMENTATION



The image shows a smartphone displaying a Visual Studio Code (VS Code) interface. The terminal window is active, showing Python code and training logs for a neural network model. The code includes imports, function definitions, and a main loop for training. The logs provide detailed information about the model's architecture, parameter counts, and training progress, including loss and accuracy metrics. The VS Code interface also shows the file structure of the project, including files like model.py, main1.py, and main2.py.

```
lization)

block_16_expand_relu (ReLU)      (None, 7, 7, 960)  0      ['block_16_expand_BN[0][0]']

block_16_depthwise (DepthwiseC (None, 7, 7, 960)  8640    ['block_16_expand_relu[0][0]']

    orn2D)

block_16_depthwise_BN (BatchNo (None, 7, 7, 960)  3840    ['block_16_depthwise[0][0]']

    rmalization)

block_16_depthwise_relu (ReLU)  (None, 7, 7, 960)  0      ['block_16_depthwise_BN[0][0]']

block_16_project (Conv2D)       (None, 7, 7, 320)   307200  ['block_16_depthwise_relu[0][0]']

block_16_project_BN (BatchNorm (None, 7, 7, 320)   1280    ['block_16_project[0][0]']

    alization)

Conv_1 (Conv2D)                (None, 7, 7, 1280)  409600  ['block_16_project_BN[0][0]']

Conv_1_bn (BatchNormalization) (None, 7, 7, 1280)  5120    ['Conv_1[0][0]']

out_relu (ReLU)                (None, 7, 7, 1280)  0      ['Conv_1_bn[0][0]']

=====
Total params: 2,257,984
Trainable params: 2,223,872
Non-trainable params: 34,112

2022-02-12 15:56:27.000256: I tensorflow/stream_executor/cuda/cuda_dnn.cc:368] Loaded cuDNN version 8200
2022-02-12 15:56:36.964663: I tensorflow/stream_executor/cuda/cuda_blas.cc:1786] TensorFloat-32 will be used for the matrix multiplication. This will only be logged once.
E:\ON\Hackathon\venv\lib\site-packages\keras\optimizer_v2\adam.py:105: UserWarning: The 'lr' argument is deprecated, use 'learning_rate' instead.
    super(Adam, self).__init__(name, **kwargs)
Epoch 1/5
E:\ON\Hackathon\venv\lib\site-packages\tensorflow\python\util\dispatch.py:1082: UserWarning: ``categorical_crossentropy`` received ``from_logits=True`` , but the ``output`` argument was produced by a sigmoid or softmax activation and thus does not represent logits. Was this intended?
    return dispatch_target(*args, **kwargs)
11/11 [=====] - 18s 1s/step - loss: 9.3664 - accuracy: 0.2388 - val_loss: 8.7581 - val_accuracy: 0.1429
Epoch 2/5
11/11 [=====] - 12s 1s/step - loss: 10.0845 - accuracy: 0.1331 - val_loss: 9.0336 - val_accuracy: 0.0714
Epoch 3/5
11/11 [=====] - 11s 951ms/step - loss: 10.8388 - accuracy: 0.0976 - val_loss: 9.0015 - val_accuracy: 0.0714
Epoch 4/5
6/11 [=====] - ETA: 4s - loss: 11.7843 - accuracy: 0.1250
```

```
1 import turtle
2 import time
3 wn=turtle.Screen()
4 wn.title("Team Project Phoenix")
5 wn.bgcolor("black")
6
7 #Draw a box around the light
8 pen=turtle.Turtle()
9 pen.color("yellow")
10 pen.width(3)
11 pen.hideturtle()
12 pen.penup()
13 pen.goto(-30,60)
14 pen.pendown()
15 pen.fd(60)
16 pen.rt(90)
17 pen.fd(120)
18 pen.rt(90)
19 pen.fd(60)
20 pen.rt(90)
21 pen.fd(120)
22
23 #Red Light
24 red_light=turtle.Turtle()
25 red_light.shape("circle")
26 red_light.color("grey")
27 red_light.penup()
28 red_light.goto(0,40)
29
30 #Yellow Light
31 yellow_light=turtle.Turtle()
32 yellow_light.shape("circle")
33 yellow_light.color("grey")
```

File Edit Selection View Go Run Terminal Help

modelpy 1 - Visual Studio Code

EXPLORER

project > model.py > ...

75 IMG_SIZE = (224, 224)
76 IMG_SHAPE = IMG_SIZE + (3,)
77 base_model = tf.keras.applications.MobileNetV2(input_shape=IMG_SHAPE,
80 include_top=False,
weights='imagenet')

81 base_model.summary()
82 len(base_model.layers)

85 image_batch, label_batch = next(iter(train_generator))
88 feature_batch = base_model(image_batch)

91 base_model.trainable = False

95 model = tf.keras.Sequential([
96 base_model,
97 tf.keras.layers.GlobalAveragePooling2D(),
98 tf.keras.layers.Dropout(0.2),
99 tf.keras.layers.Dense(256, activation='relu'),
100 tf.keras.layers.Dropout(0.2),
101 tf.keras.layers.Dense(5, activation='softmax')
102])
103
104
105
106
107 base_learning_rate = 0.0001
108 model.compile(optimizer=tf.keras.optimizers.Adam(lr=base_learning_rate),
109 loss=tf.keras.losses.CategoricalCrossentropy(from_logits=True),
110 metrics=['accuracy'])
111
112 initial_epochs = 5

113



File Edit Selection View Go Run Terminal Help

modelpy 1 - Visual Studio Code

EXPLORER

project > model.py > ...

38 train_generator=train_data.flow_from_directory(
39 traindir,
40 target_size=(224, 224),
41 class_mode='categorical',
42 batch_size=32
43)
44
45 test_generator=test_data.flow_from_directory(
46 testdir,
47 target_size=(224, 224),
48 class_mode='categorical',
49 batch_size=32
50)
51
52 validation_generator=test_data.flow_from_directory(
53 valdir,
54 target_size=(224, 224),
55 class_mode='categorical',
56 batch_size=32
57)
58
59 class_weights = class_weight.compute_class_weight('balanced', np.unique(train_generator.classes),train_generator.classes)
60
61 target_labels = next(os.walk(traindir))[1]
62
63 target_labels.sort()
64
65 batch = next(train_generator)
66 batch_images = np.array(batch[0])
67 batch_labels = np.array(batch[1])
68
69 target_labels = np.asarray(target_labels)
70
71
72
73
74
75 IMG_SIZE = (224, 224)
76 IMG_SHAPE = IMG_SIZE + (3,)
77 base_model = tf.keras.applications.MobileNetV2(input_shape=IMG_SHAPE,

Python 3.9.7 ('env: venv') 0 1
Type here to search
21°C Haze
Powered by 000webhost

BUSINESS MODEL

Niche market; not much competition

Highly cheaper than the competition

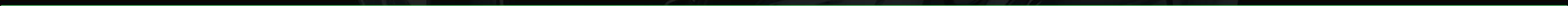
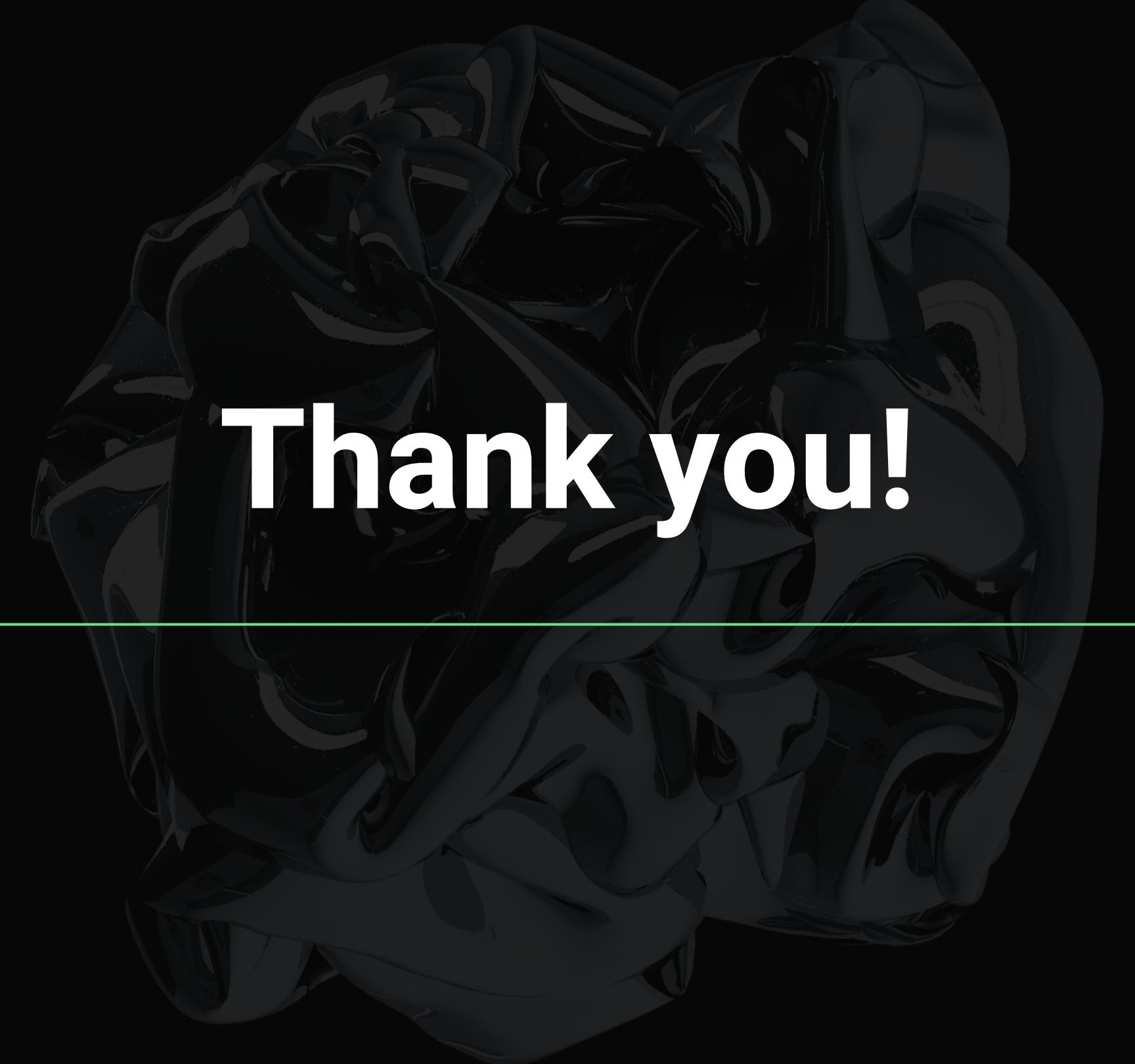
Continuous investment in security

Complex event streaming combined with data management capabilities, also known as anomaly detection, can be instrumental to effectively verify the urgency of events in real time and facilitate predictive interventions.

“

Just a few seconds
can mean the difference between
life and death

CRAIG STOUT



Thank you!