inovex

Let's talk about Computer Vision!

Silvia Santano and Pepper

Köln, 19.02.2018

About Me

- Silvia Santano
- Pepper Applications development
- At inovex since June 2016
- Programming robots since I was 12

Agenda

- Computer Vision
- Image Recognition
- Pepper
  - Characteristics
  - Computer Vision with Pepper
  - External services
    - Google
    - Microsoft
  - On-device CV with CNNs

Agenda

- Computer Vision
- Image Recognition
- Pepper
  - Characteristics
  - Computer Vision with Pepper
  - External services
    - Google
    - Microsoft
  - On-device CV with CNNs

Computer Vision

Automatic extraction, analysis and understanding of information from images

Image: http://www.puretechsystems.com/images/ComputerVision3.jpg

Image: https://fineartamerica.com/featured/finlayson-old-house-susan-crossman-buscho.html

Image: http://www.cavareno.com/how-to-be-green-at-home/best-green-home-design-138869-2/.jpg

Image: http://wallpaperose.com/house-in-the-forest-painting.html

Image: http://www.dezineguide.com/inspiration/15-creative-exterior-houses-designs-examples/.jpg

Humans can recognize objects in images
with little effort despite of huge variations

For computers this is still a challenge…

Agenda

- Computer Vision
- Image Recognition
- Pepper
  - Characteristics
  - Computer Vision with Pepper
  - External services
    - Google
    - Microsoft
  - On-device CV with CNNs

Image Recognition

Determine whether or not an image contains a specific object

Image Recognition

Main subfields:

- **Classification**
- Object detection
- Semantic segmentation
- Identification

http://www.coldvision.io/wp-content/uploads/2016/07/dnn_classification_vs_detection.png

Image Recognition

Main subfields:

- Classification
- **Object detection**
- Semantic segmentation
- Identification

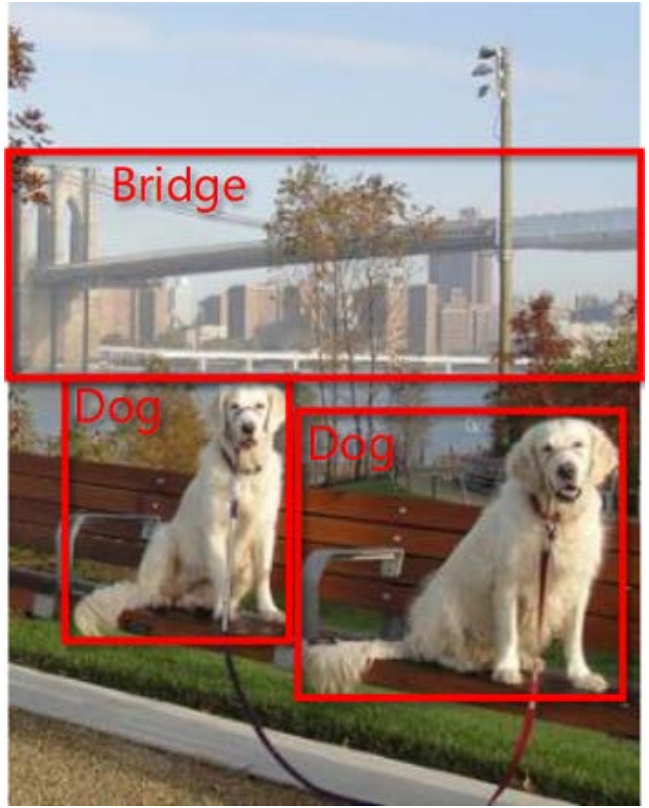
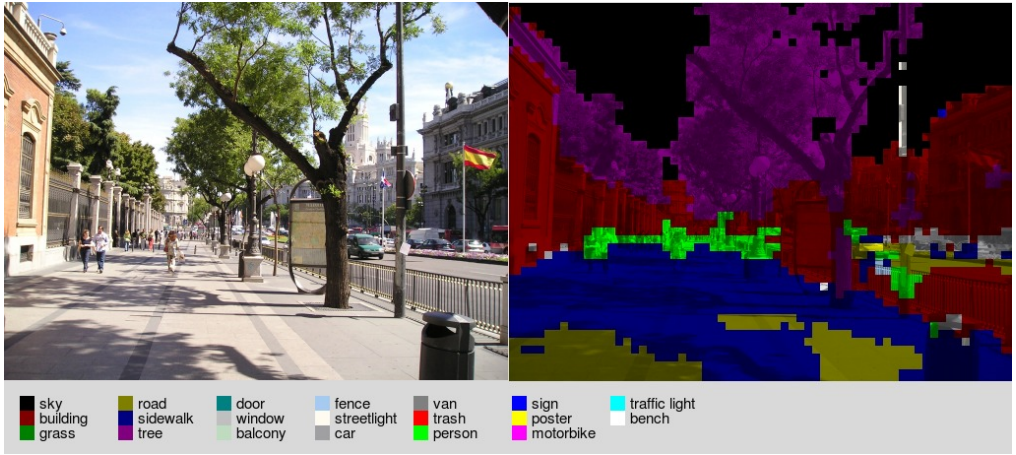


Image Recognition

Main subfields:

- Classification
- Object detection
- **Semantic segmentation**
- Identification

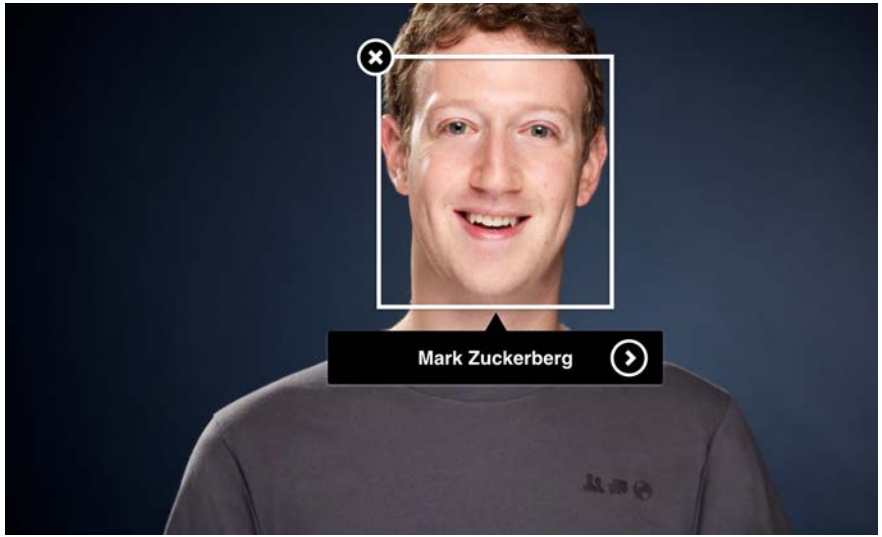


Images: http://david.grangier.info/scene_parsing/

Image Recognition

Main subfields:

- Classification
- Object detection
- Semantic segmentation
- **Identification**

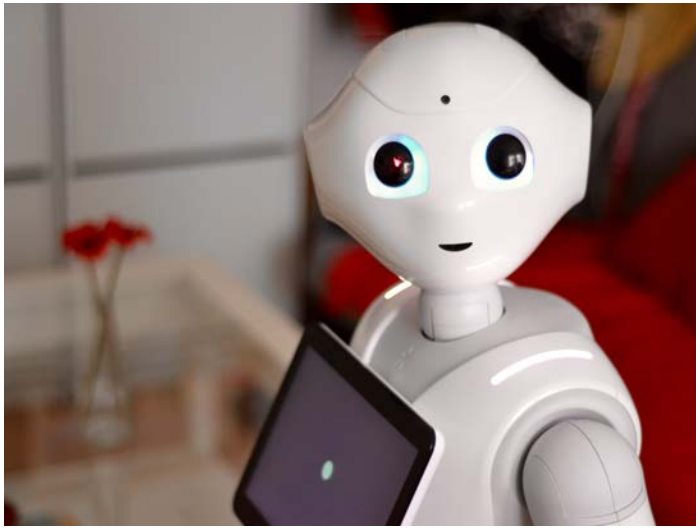


Agenda

- Computer Vision
- Image Recognition
- Pepper
  - Characteristics
  - Computer Vision with Pepper
  - External services
    - Google
    - Microsoft
  - On-device CV with CNNs

Pepper: Capabilities

- Softbank Robotics
- Voice and gestures
- Face and emotion recognition
- Internet Connection
- Tablet
- Safety mechanisms, automatic balance, and anti-colission system



Pepper: Technical Characteristics (v1.8A)

PROCESSOR	Atom E3845
CPU	Quad core
Clock speed	1.91 GHz
RAM	4 GB DDR3
OS	Nao QI OS
2 HD Cameras (OV5640) 1 3D Sensor (ASUS XTION) 4 Microphones A 3-axis Gyrometer and a 3-axis Accelerometer 6 laser line generators 2 Infra-Red sensors 2 ultrasonic sensors 3 tactile sensors 3 bumpers 20 Motors and actuators	

Tablet

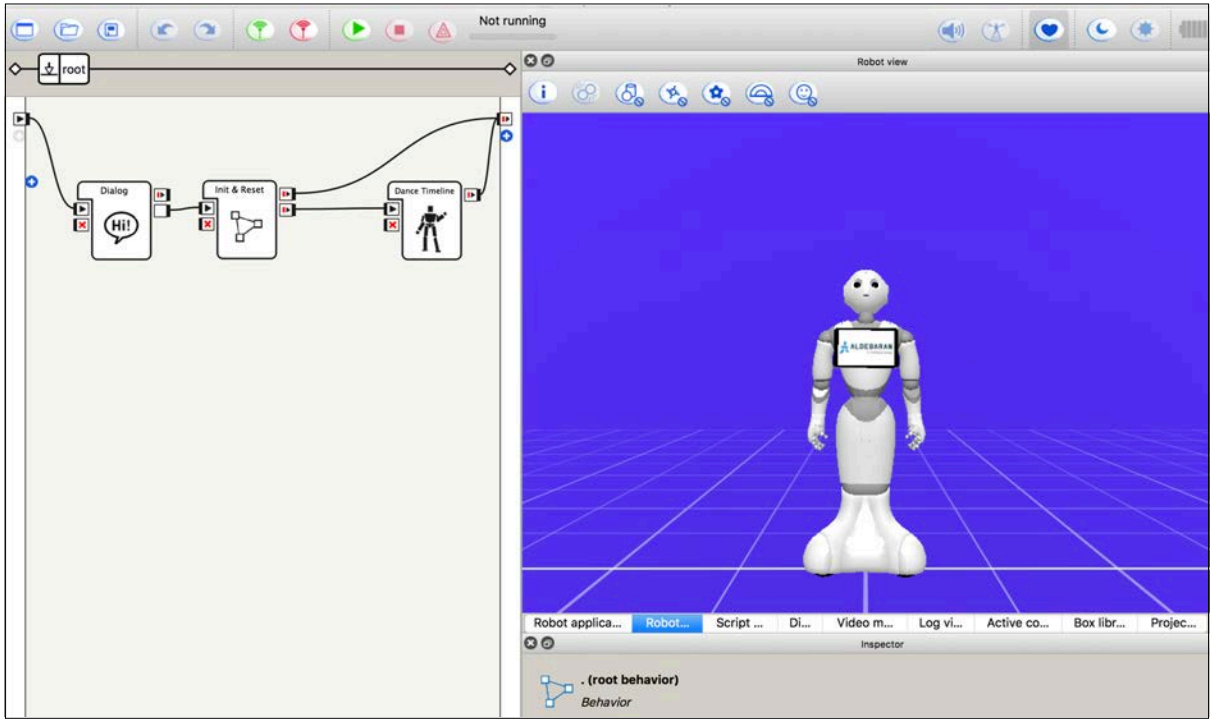
Dimensions	246 x 175 x 14.5 mm
CPU	1.3 GHz quad-core ARM Cortex-A7 Cache 512 KB L2 Wi-Fi, Bluetooth 1.6G pixel/sec @416MHz
DDR3 SDRAM	1GB (512MB * 2)
Flash Memory	32GB (eMMC)
Display	Type: IPS, Resolution: 1280*800 Color: 24bit true color
Touch Panel	Capacitive Multi-Touch (5 point)
Sensors	Light illumination, Acceleration Gyro, Geomagnetic
OS	Android (6.0)

Programming Pepper

- Choregraphe und Python
- Python
- C++
- Javascript
- Soon: Android (reduced function set)

Programming Pepper

Choregraphe



Programming Dialogs
QiChat

```
# Volume
##down
u:([
    "{~can_you} {"ein bisschen" etwas} leiser [sprechen reden]"
    "sprich {"ein bisschen" etwas} leiser"
    "Dreh die Lautstärke runter"
    "sprich nicht so laut"
    "du sprichst zu laut"
])
^gotoReactivate(decrease_volume)
u:($empty) %decrease_volume
^call(ALVolumeSlider.decreaseVolume()) $Demo/back=1
    c1:(false) es tut mir leid, das ist das Minimum
    c1:(true) okay ich spreche jetzt leiser

        u2:([
            nochmal
            mehr
            "noch {"ein bisschen" etwas} mehr"
            "immer noch zu laut"
            ])
            ^gotoReactivate(decrease_volume)
```

Agenda

- Computer Vision
- Image Recognition
- Pepper
  - Characteristics
  - Computer Vision with Pepper
  - External services
    - Google
    - Microsoft
  - On-device CV with CNNs

Computer Vision with Pepper

- Face Detection and People Tracking
- Face Learning and Recognition
- People Characteristics Perception

Computer Vision with Pepper

People Characteristics Perception

```
PeoplePerception/Person/<ID>/AgeProperties
PeoplePerception/Person/<ID>/ExpressionProperties
PeoplePerception/Person/<ID>/GenderProperties
PeoplePerception/Person/<ID>/SmileProperties
PeoplePerception/Person/<ID>/FacialPartsProperties
PeoplePerception/Person/<ID>/Distance
PeoplePerception/Person/<ID>/IsFaceDetected
PeoplePerception/Person/<ID>/IsVisible
PeoplePerception/Person/<ID>/NotSeenSince
PeoplePerception/Person/<ID>/PresentSince
PeoplePerception/Person/<ID>/RealHeight
PeoplePerception/Person/<ID>/ShirtColor
```

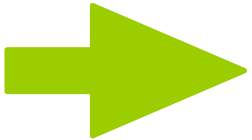
Computer Vision with Pepper

- Face Detection and People Tracking
- Face Learning and Recognition
- People Characteristics Perception
- Emotion Recognition

Computer Vision with Pepper

Emotion Recognition Module

- Data sources:
  - Expression and smile
  - Acoustic voice emotion analysis
  - Head angles
  - Touch sensors
  - Semantic analysis from speech
  - Sound level and energy level of noise
  - Movement detection



```
Valence
Attention Level
Smile
Expression
{
     “calm"
     “anger"
     “joy"
     "sorrow"
     "laughter"
     "excitement"
     "surprise"
}
(Real values normalized)
```

Computer Vision with Pepper

- People Tracking
- Face Detection, Learning and Recognition
- People Perception
- Emotion Recognition
- Vision Recognition
- Barcode Reader

Computer Vision with Pepper

DEMO:

People Perception

Emotion Recognition

Face Detection and Recognition

Agenda

- Computer Vision
- Image Recognition
- Pepper
  - Characteristics
  - Computer Vision with Pepper
  - External services
    - Google
    - Microsoft
  - On-device CV with CNNs

External services integration

Google's Machine Learning Cloud Vision API

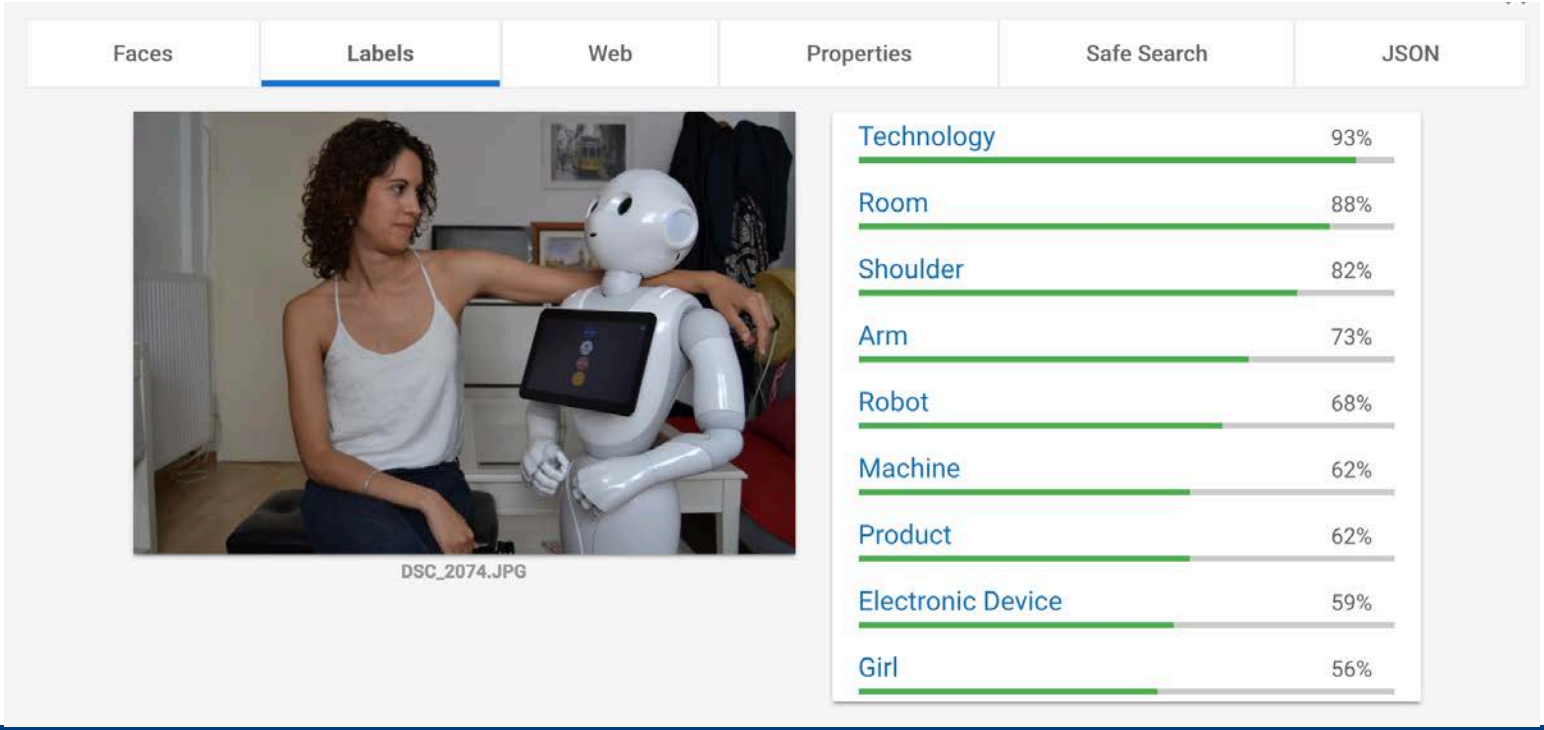
- Machine learning service with pre-trained models
- JSON REST API + client libraries (C#, GO, Java, Node.js, PHP, Python, Ruby)

Explicit Content Detection
Logo Detection
Label Detection
Landmark Detection
Optical Character Recognition
Face Detection
Image Attributes
Web Detection

Google Cloud Platform

External services integration

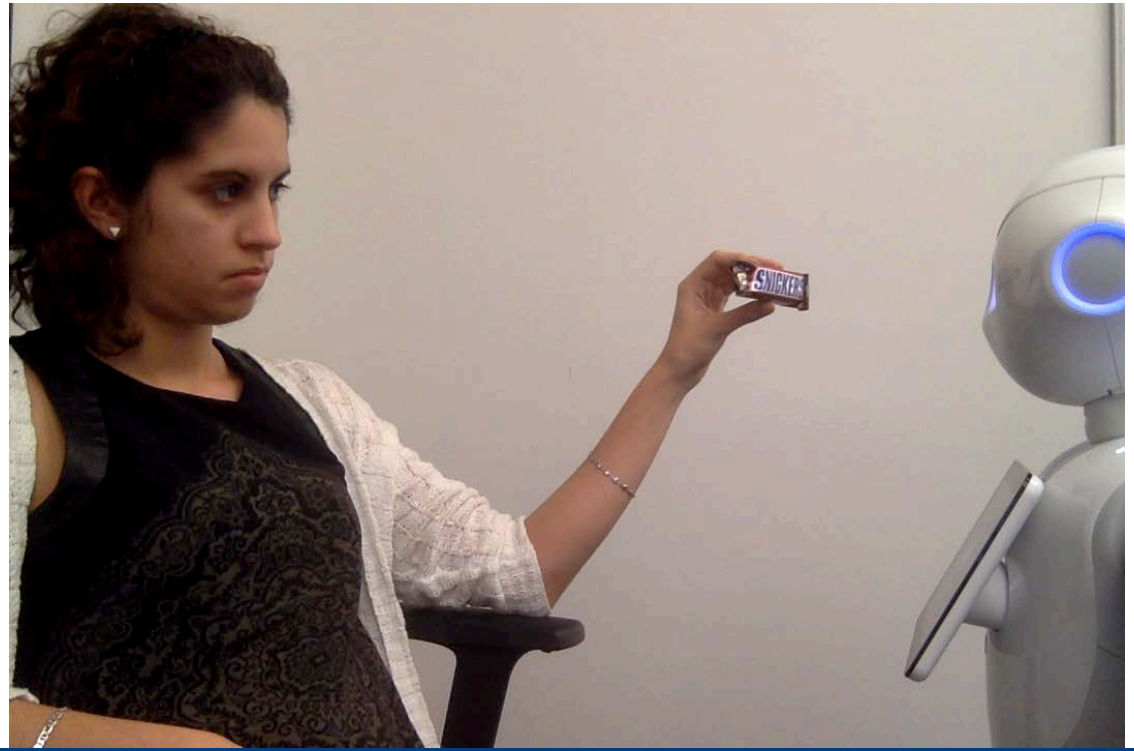
Google's Machine Learning Cloud Vision API: LABELS



https://cloud.google.com/vision/

External services integration

Google's Machine Learning Cloud Vision API: LOGO DETECTION



External services integration

Google's Machine Learning Cloud Vision API: LABELS

Show me the code

External services integration

Google's Machine Learning Cloud Vision API: LABELS

```python
def detect_labels(path):
    """Detects labels in the file."""
    client = vision.ImageAnnotatorClient()

    with io.open(path, 'rb') as image_file:
        content = image_file.read()

    image = types.Image(content=content)

    response = client.label_detection(image=image)
    labels = response.label_annotations
    print('Labels:')

    for label in labels:
        print(label.description)
```

client libraries (C#, GO, Java, Node.js, PHP, **Python**, Ruby)

External services integration

Google's Machine Learning Cloud Vision API: LABELS

```
POST https://vision.googleapis.com/v1/images:annotate?key=YOUR_API_KEY
```

```
{
  "requests": [
    {
      "image": {
        "content": "/9j/7QBEUGhvdG9zaG9...base64-encoded-image-content...fXNWzvDE
      },
      "features": [
        {
          "type": "LABEL_DETECTION"
        }
      ]
    }
  ]
}
```

JSON REST API

External services integration

Google's Machine Learning Cloud Vision API

DEMO:

Logo Detection

Label Detection

Optical Character Recognition

Web Detection

Emotion Detection

External services integration

Microsoft Cognitive Services

- Machine learning service with pre-trained models
- JSON REST APIS + client libraries (C#, Android, Swift)

Computer Vision API:
- Analyze Image
- Optical Character Recognition
- Handwritten Text Detection

Face API

Emotion API

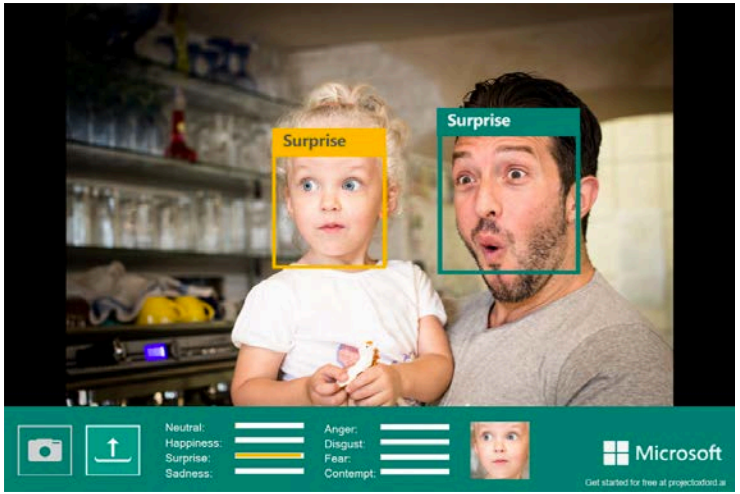
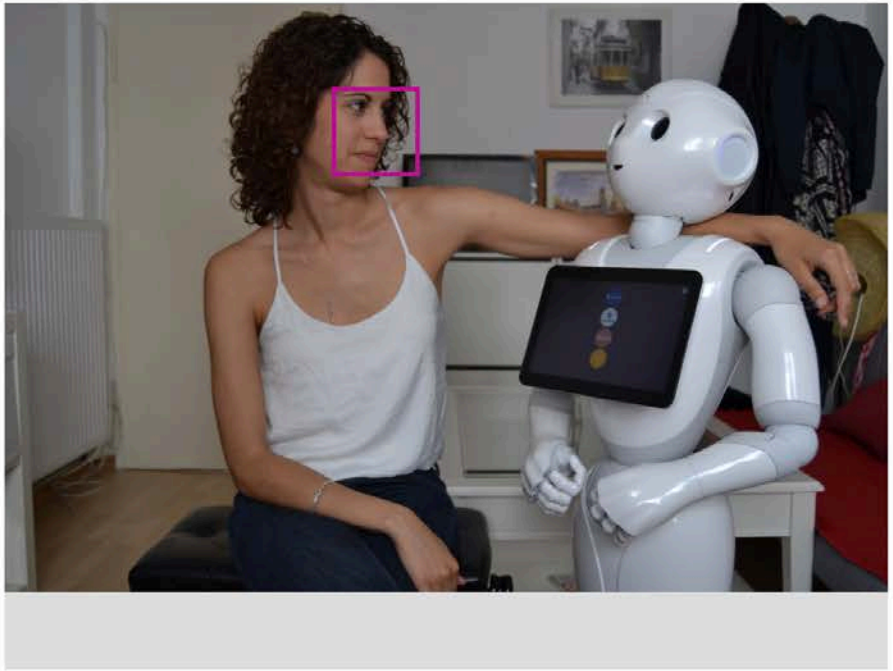


Image: blogs.microsoft.com

External services integration

Microsoft Cognitive Services: COMPUTER VISION API

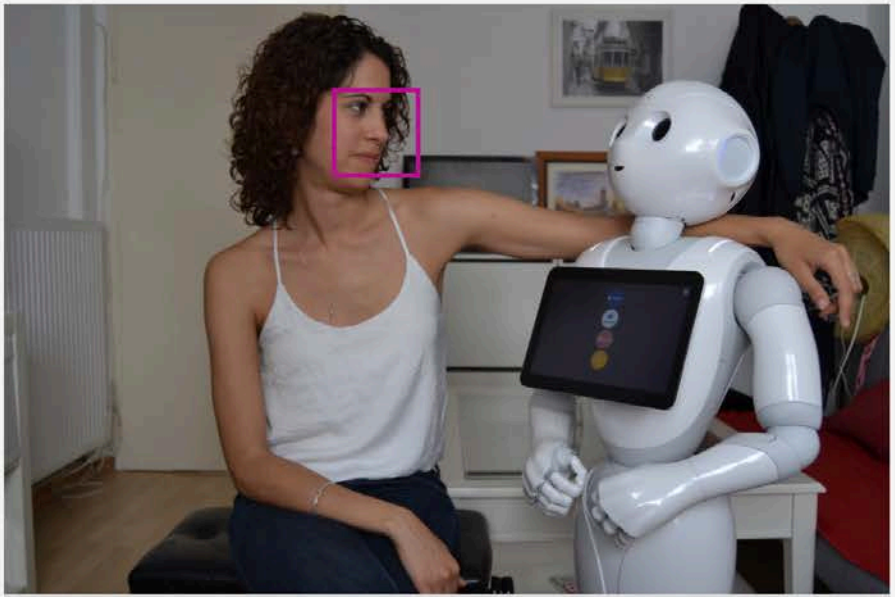


FEATURE NAME:	VALUE
Description	{ "tags": ["person", "indoor", "woman", "holding", "man", "front", "table", "white", "black", "standing", "young", "cake", "playing", "dog", "room", "plate", "remote", "kitchen"], "captions": [{ "text": "a woman standing in front of a cake", "confidence": 0.75387466 }] }
Tags	[{ "name": "wall", "confidence": 0.996851742 }, { "name": "person", "confidence": 0.996797 }, { "name": "indoor", "confidence": 0.973828435 }]
Image format	"Jpeg"
Image dimensions	1080 x 1620

https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision/

External services integration

Microsoft Cognitive Services: FACE API




```
"gender": "female",
"age": 29.4,
"facialHair": {
  "moustache": 0.0,
  "beard": 0.0,
  "sideburns": 0.0
},
"glasses": "NoGlasses",
"makeup": {
  "eyeMakeup": true,
  "lipMakeup": true
},
"emotion": {
  "anger": 0.0,
  "contempt": 0.001,
  "disgust": 0.0,
  "fear": 0.0,
  "happiness": 0.01,
  "neutral": 0.973,
  "sadness": 0.015,
  "surprise": 0.0
},
"occlusion": {
```

External services integration

Microsoft Cognitive Services

DEMO:

Analyze Image

Optical Character Recognition

Handwritten Text Recognition

Emotion Detection

Agenda

- Computer Vision
- Image Recognition
- Pepper
  - Characteristics
  - Computer Vision with Pepper
  - External services
    - Google
    - Microsoft
  - On-device CV with CNNs

On-device CV with CNNs

Why

- Privacy
- Latency
- Connectivity
- Security
- Cost

On-device CV with CNNs

Limitations

- Compute
- Memory
- Storage
- Power
- Bandwidth

On-device CV with CNNs

Tools

- e.g. Tensorflow Mobile
- or Tensorflow Lite

Pre-trained models

Tensorflow Object Detection API

TensorFlow

Image: softbank.jp

…Out of curiosity



Google's Algorithm found houses


Google's Algorithm found no houses

Yes, but: chihuahua or muffin?

Comparison

- **Amazon's Rekognition** is not just good at identifying the primary object but also the many objects around it
- **Google's Vision API** and **IBM Watson Vision** return straightforward, descriptive labels
- **Microsoft's** tags were usually too high level
- **Cloudsight** is a hybrid between human tagging and machine labelling. More accurate. Slower. More expensive.
- **Clarifai** returns, by far, the most tags (at 20) although very generic tags. It also adds qualitative and subjective labels, such as "cute", "funny", "adorable", and "delicious"

https://medium.freecodecamp.org/chihuahua-or-muffin-my-search-for-the-best-computer-vision-api-cbda4d6b425d

Vielen Dank

Silvia Santano
Application Development

@SilviaSantano
linkedin.com/in/silviasantano
ssantano@inovex.de
0173 3181 085