T-SYSTEMS MULTIMEDIA SOLUTIONS

Getting started with Turi Create and CoreML

Apple's machine learning pipeline for iOS

DIGILITY, Cologne – Volker Bublitz – 26.09.2018

DIGITAL SERVICE & MOBILE

Allow me, MMS!

23 **YEARS**
Digital Business

1.900

Digital Natives, Experts, Unconventionel Thinkers, Project Enthusiast, Consultant, Developers, Nerds, Controllers, Managers, Sales Representatives, Business Whisperers, Testers and Innovators

Overall Customer Satisfaction 2017: **97%**

Average Age: **35 YEARS**

Work Culture: agile, flexible, professional, reliable, digital

Total Numbers of Customer Projects in 2017: **2.782**

DIGITAL SERVICE & MOBILE

MOBILE EXPERTISE

> 65 

Colleagues working in the mobile space
Project Managers, Architects, Designers, Testers, Developers and Consultants

PROFESSIONAL SERVICES

- Strategy and Process Consulting
- Project Management
- Design and Technical Conception
- Design Consulting and Prototyping
- Research and Development
- Feature and Performance Testing
- Controlling
- System and App Monitoring
- Mobile Core Checking

Single Source and Contact
From Back to Front End

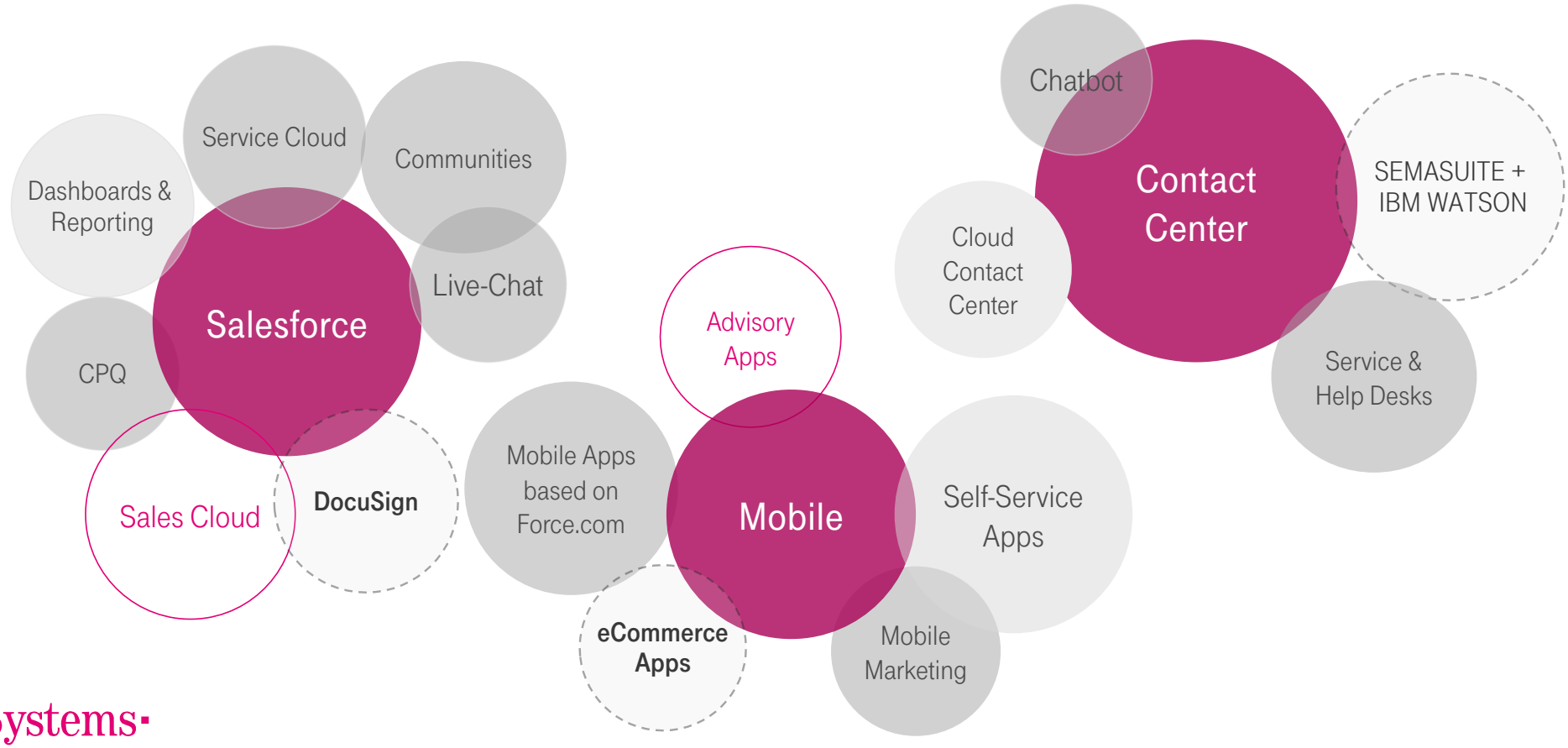
Project Methodology
Agile methods - SCRUM and KANBAN - but also traditional iterative Waterfall Modell

Certified Testing Lab

T-SYSTEMS MULTIMEDIA SOLUTIONS

CORE TOPICS OF DIGITAL SERVICES@MMS

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS



T · · Systems ·

Getting started with Turi Create and CoreML

Apple's machine learning pipeline for iOS

T · · Systems ·

THE TRAINER

ABOUT TURI CREATE

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

What Apple says...

- Python library for creating Core ML models
- Easy to use, no need to be an ML expert

„You don't have to be a machine learning expert to add recommendations, object detection, image classification, image similarity or activity classification to your app.“

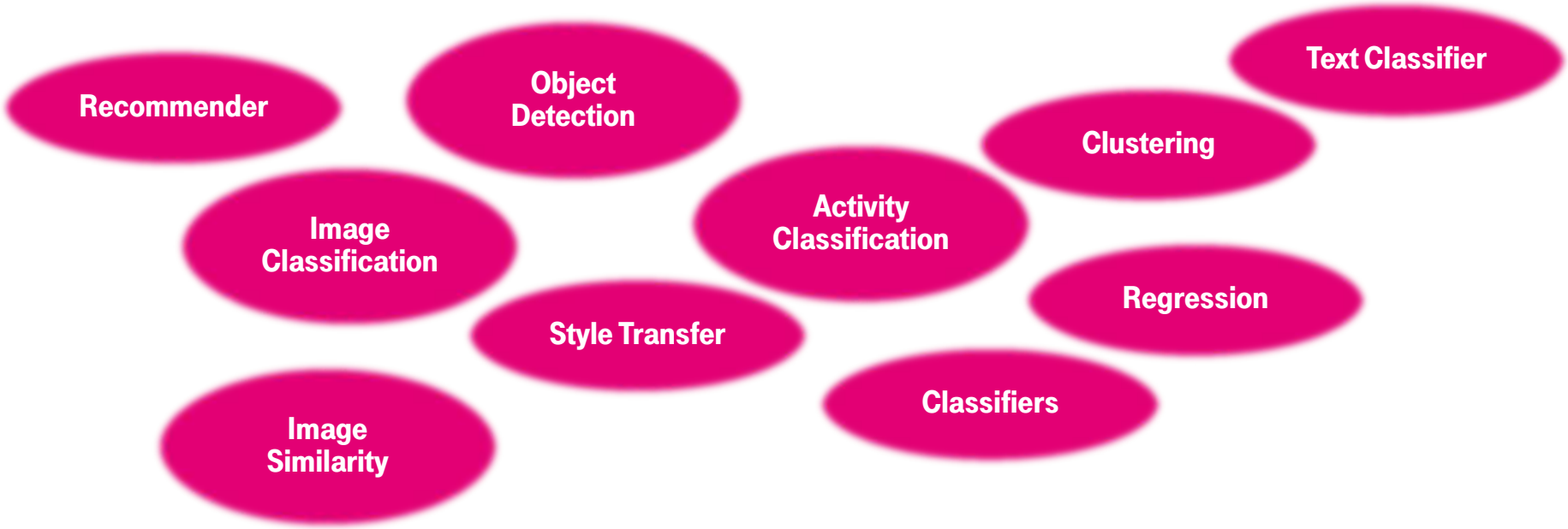
Source: https://github.com/apple/turicreate

T · · Systems ·

ABOUT TURI CREATE

SUPPORTED ML TASKS

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS



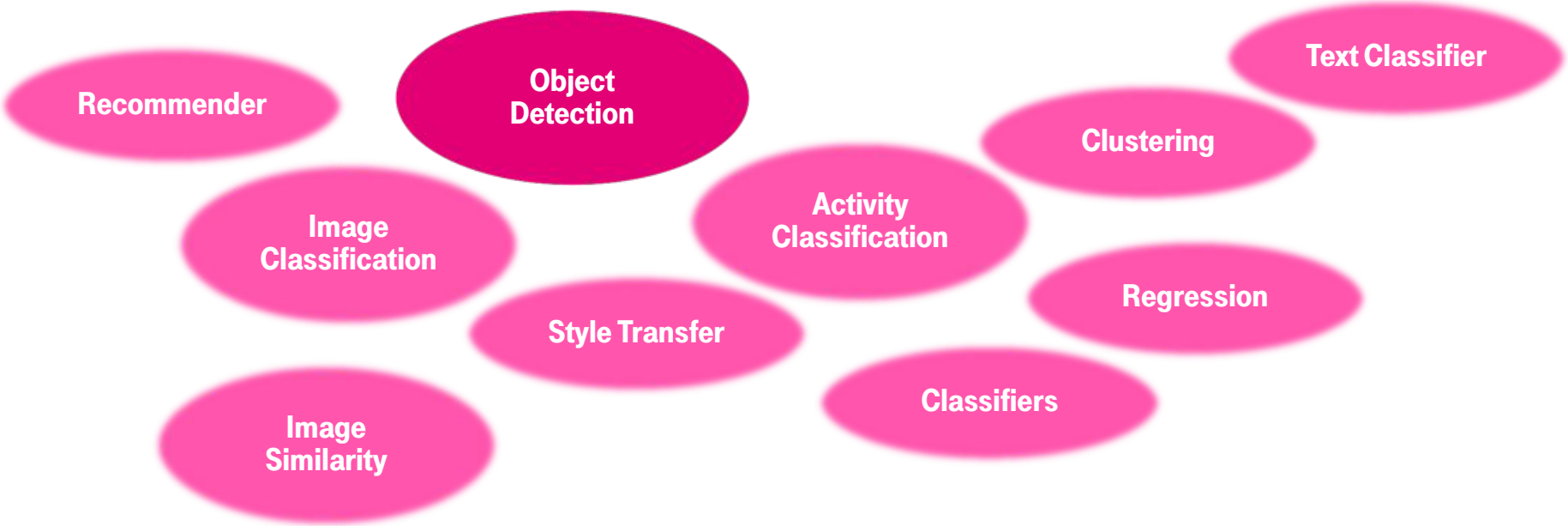
Source: https://github.com/apple/turicreate

T · · Systems ·

ABOUT TURI CREATE

SUPPORTED ML TASKS

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS



Source: https://github.com/apple/turicreate

T · · Systems ·

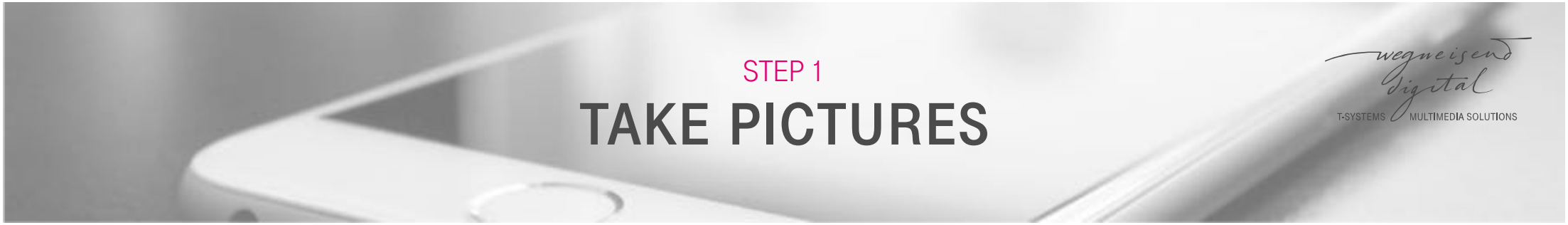
THE MISSION

OBJECT DETECTION IN 4 STEPS

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

1. TAKE PICTURES
2. ANNOTATE IMAGES
3. TRAIN A MODEL
4. USE THE MODEL IN YOUR APP

T · · Systems ·



STEP 1

TAKE PICTURES

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

For a simple app ...

1. Take 40-60 pictures for each single object
2. Use different angles and varying lighting

T · · Systems ·

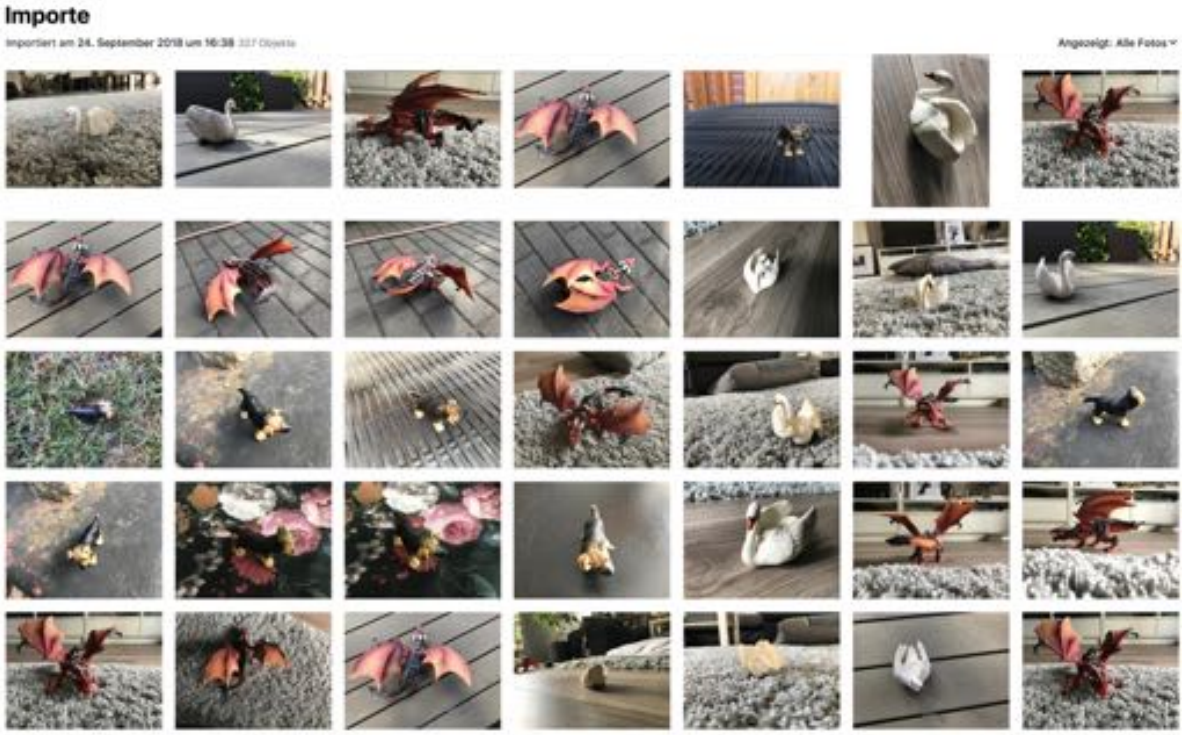
DEMO

T · · Systems ·

STEP 1

TAKE PICTURES

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS



T · · Systems ·

STEP 2

ANNOTATE IMAGES

wegweisend digital

T-SYSTEMS MULTIMEDIA SOLUTIONS

What Turi Create can work with...

T · · Systems ·

STEP 2

ANNOTATE IMAGES

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

Simple JSON:

```
{
  "path" : [   "images\/IMG_2603.jpg",   "images\/IMG_2617.jpg", ...],
  „annotations“: [ ... ]
}
```

T · · Systems ·

STEP 2

ANNOTATE IMAGES

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

JSON annotation data ...

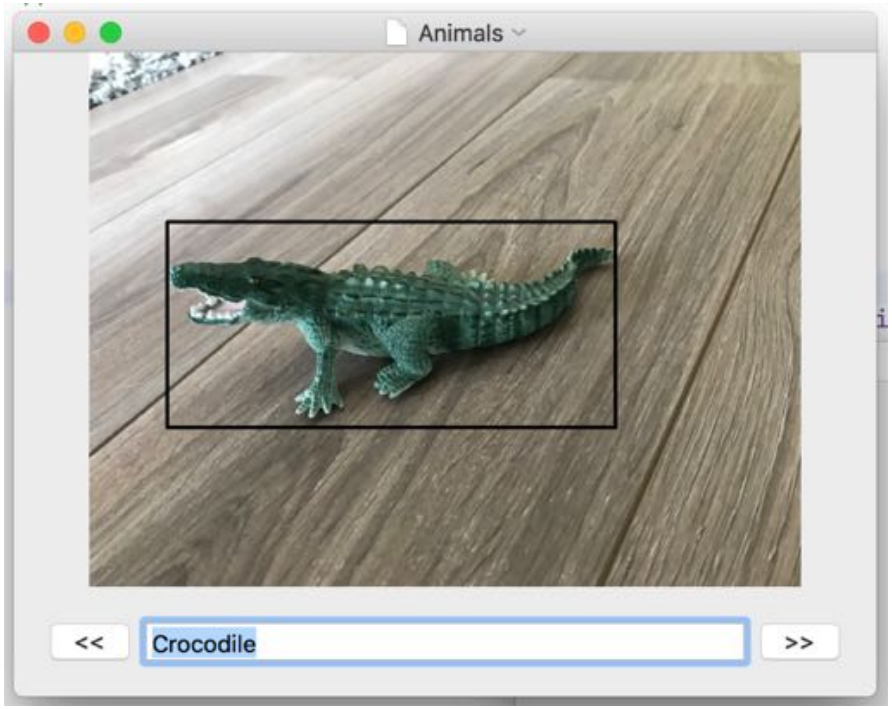
```
{
  "label" : "Crocodile",
  "coordinates" : {
    "x" : 567.99, "y" : 529.88,
    "width" : 856.79, "height" : 344.42 }
}
```

T · · Systems ·

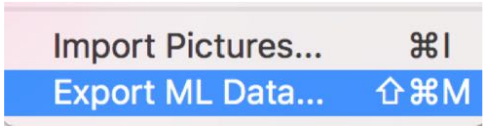
STEP 2

ANNOTATE IMAGES

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS



Use a tool for this!



T · · Systems ·

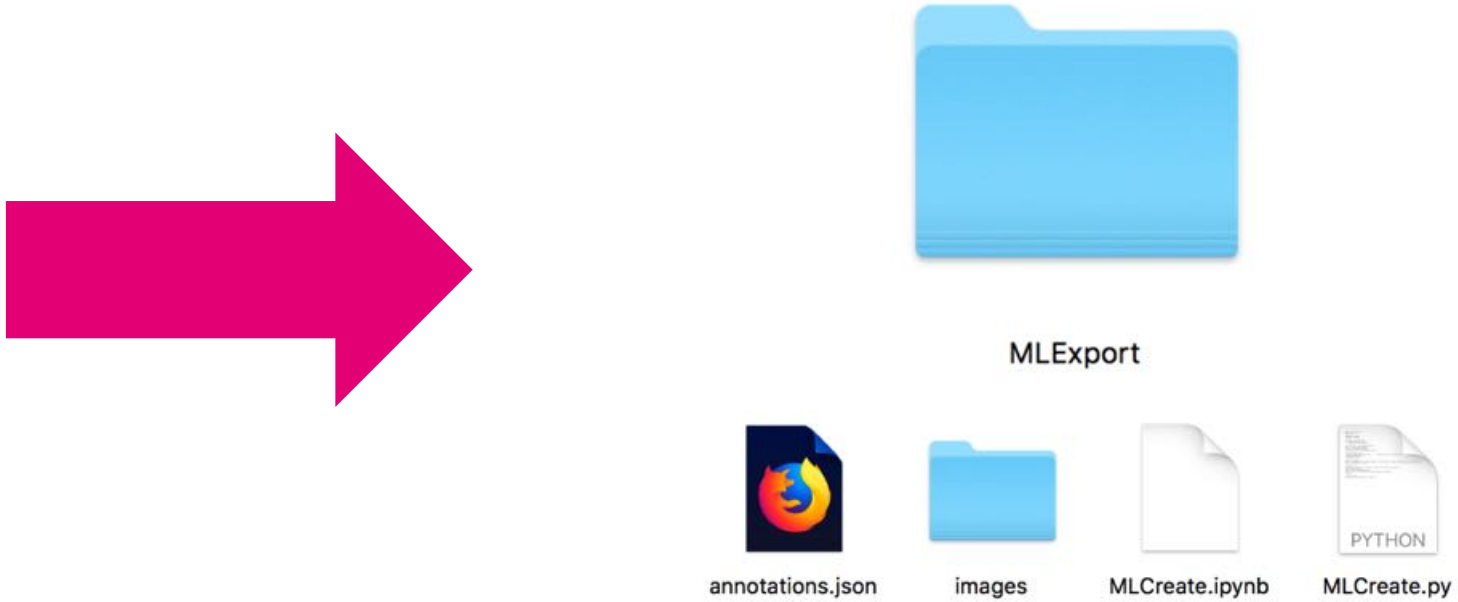
DEMO

T · · Systems ·

STEP 2

ANNOTATE IMAGES

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS



T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

It's as simple as this...

```
import turicreate as tc
data = tc.SFrame('mymodel.sframe')  # Load data
model = tc.object_detector.create(data)  # Create a model
predictions = model.predict(data)  # Make predictions
model.export_coreml('MyClassifier.mlmodel') # Export to Core ML
```

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

But...

```
# Load data
data = tc.SFrame(mymodel.sframe')
```

„sframe“? We have a folder of images and JSON???

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

```
# Create a model

model = tc.object_detector.create(data)
```

But is everything set up correctly?

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

```
# Make predictions
predictions = model.predict(data)
```

Don't we want separate test data for this task?

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

```
# Export to Core ML
model.export_coreml('MyClassifier.mlmodel')
```

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

```
# Export to Core ML
model.export_coreml('MyClassifier.mlmodel')
```

Ok!

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

Of course you can handle this easily... Load your data...

```
with open('annotations.json') as j:
    annotations = json.load(j)
annotationData = tc.SFrame(annotations)
data = tc.load_images('images/')
data = data.join(annotationData)
```

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

Create test data...

```
trainData, testData = data.random_split(0.8)
```

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

Check if everything is set up correctly...

```
trainData['image_with_ground_truth'] = \
    tc.object_detector.util.draw_bounding_boxes(trainData['image'],
          trainData['annotations'])
trainData.explore()
```

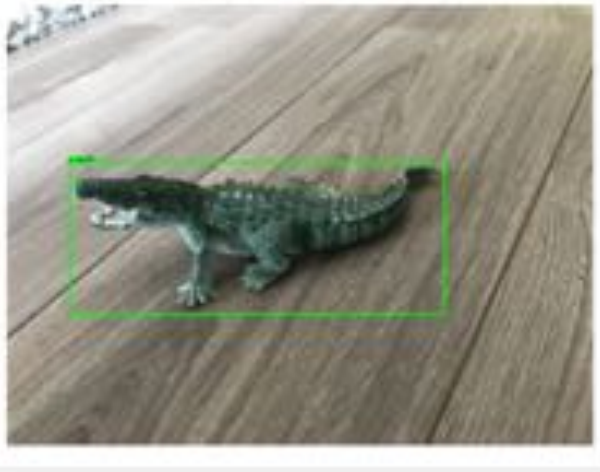
T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

annotations	image_with_ground_truth
{"label":"Crocodile","coordinates...	
{"label":"Crocodile","coordinates...	
{"label":"Bear","coordinates":{"y...	
{"label":"Bear","coordinates":{"y...	



T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

Test with test data...

```
predictions = model.predict(testData, confidence_threshold=0.05)
predictions.explore()
metrics = model.evaluate(testData)
print('mAP: {:.1%}'.format(metrics['mean_average_precision_50']))
```

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

Build train JSON SFrame

```
In [4]: with open('annotations.json') as j:
            annotations = json.load(j)

In [5]: annotationData = tc.SFrame(annotations)

In [6]: data = tc.load_images('images/')

In [7]: data = data.join(annotationData)

In [8]: trainData, testData = data.random_split(0.8)
```

Check ground truth

```
In [9]: trainData['image_with_ground_truth'] = \
            tc.object_detector.util.draw_bounding_boxes(trainData['image'], trainDa

In [10]: trainData.explore()
```

```
Materializing SFrame
```

Train the model

```
In [11]: model = tc.object_detector.create(trainData, feature="image", annotations="
```

Build your script step by step with Jupyter Notebook

Jupyter Notebook: http://jupyter.org

T · · Systems ·

DEMO

T · · Systems ·

STEP 3

TRAIN A MODEL

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

T · · Systems ·

STEP 4

USE THE MODEL IN YOUR APP

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

```
let model = try VNCoreMLModel(for: MyModel().model)
```

T · · Systems ·

STEP 4

USE THE MODEL IN YOUR APP

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

```
let request = VNCoreMLRequest(model: model) { request, error in
    guard let results = request.results as? [VNRecognizedObjectObservation] else {
        return
    }
    results.forEach { (result) in
        let confidence = result.confidence
        // get only the first classification
        let labels = result.labels.first?.identifier
        let coordinates = result.boundingBox
            //...
        }
    }
}
```

T · · Systems ·

STEP 4

USE THE MODEL IN YOUR APP

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

```
let handler = VNImageRequestHandler(cvPixelBuffer: pixelBuffer, options: [:])
try? handler.perform([request])
```

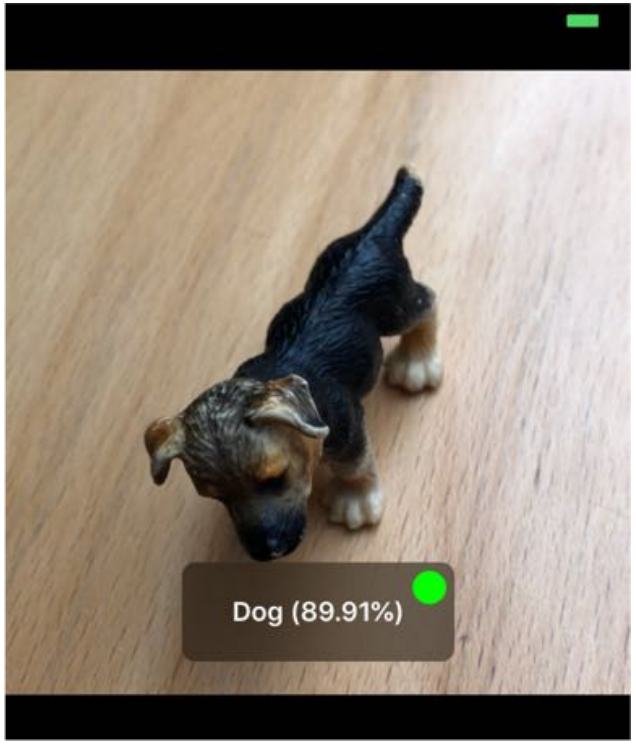
T · · Systems ·

STEP 4

USE THE MODEL IN YOUR APP

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS

MyModel.mlmodel

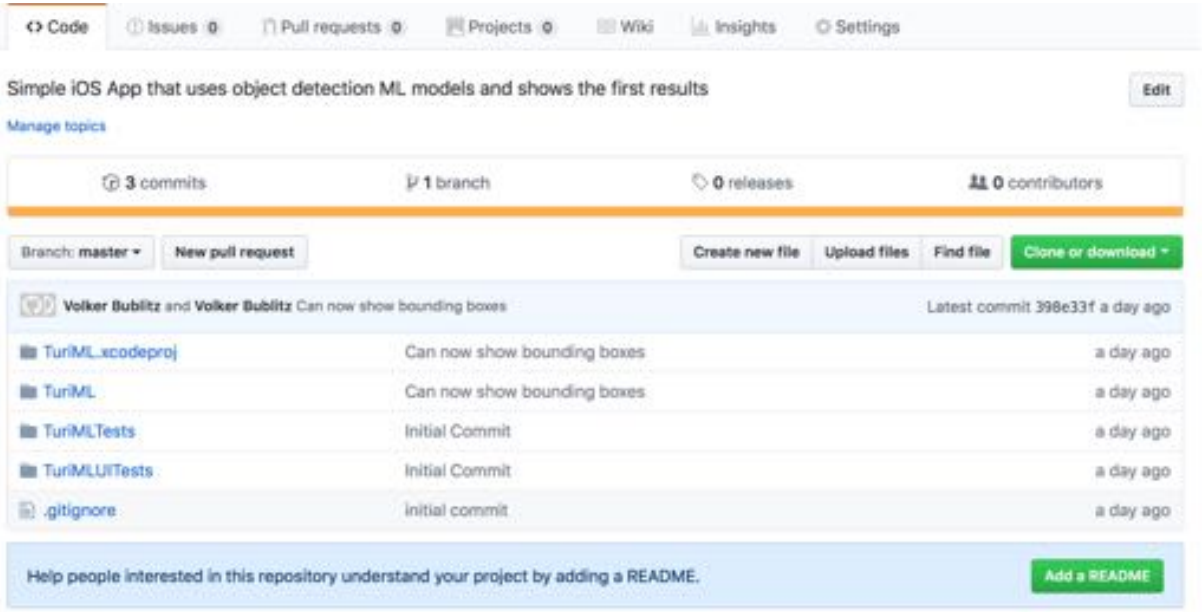


T · · Systems ·

STEP 4

USE THE MODEL IN YOUR APP

wegweisend digital
T-SYSTEMS MULTIMEDIA SOLUTIONS



TuriML.xcodeproj

T · · Systems ·

DEMO

T · · Systems ·

THANK YOU!

Volker Bublitz
volker.bublitz@t-systems.com

T · · Systems ·