#digitalshoestring

DIGITAL MANUFACTURING ON A SHOESTRING

Creating digital solutions that work for manufacturing SMEs

Hackathon

24 – 25 October 2020

digital manufacturing on a shoestring

Pitch-In paccurate ptc VOS VOCO TQC Automation & Test Solutions tqc.co.uk Leak Testing - Automation - Robotics

UNIVERSITY OF CAMBRIDGE Department of Engineering

IfM MANAGEMENT TECHNOLOGY POLICY

UKRI Engineering and Physical Sciences Research Council

University of Nottingham Institute for Advanced Manufacturing

#digitalshoestring

Agenda

Saturday 24 October

Time	Event
9:00 – 9:30	Introduction and Welcome talks
9:30 – 13:00*	**Hackathon Goal 1:** • Initial problem statement, constraints and functionality • Concept sketches and rough plan
13:00 – 18:00	**Hackathon Goal 2:** • Proof of concept with initial functionality
18:00 – 19:00	**Hackathon Goal 3:** • Plan for next day, tutorial draft, teams wrap up • End of Day 1

**Talks and Q&A from Dr Eva Agapaki (PTC) and James Malley/Pat Powers (Paccurate) will be scheduled for midday and early afternoon of Sat 24 October*

Sunday 25 October

Time	Event
9:00 – 13:00	**Hackathon Goal 4:** • Finish the prototyped system with full functionality
13:00 – 15:30	**Hackathon Goal 5:** • Validation and testing, documenting failures and mitigation → presentation, finish tutorial (ideally teams divide responsibilities)
15:30 – 18:00	**Afternoon Refreshments** **Judging and Awards** • Elevator pitch 3 minutes • 3 minutes questions per team • End of Day 2

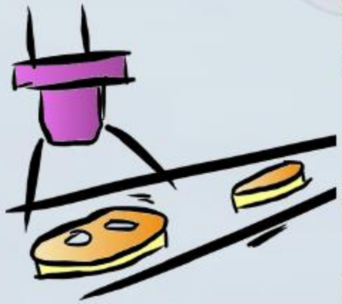
24-25 October, 2020

Automated Machine Vision Quality Inspection

Challenge:

Develop a vision system for inspecting products against predefined quality standards

Machine vision based quality inspection allows product quality to be checked and tracked in real time. In this challenge, participants will develop a 2D low-cost visual inspection system able to pass/fail parts or assemblies. The solution must be able to support different levels of inspection such as with or without realistic backgrounds. There are 3 phases to this challenge:



1) Integrate low-cost cameras, vision and image processing libraries
2) Set up, configure and deploy the system
3) Provide PASS/FAIL feedback

__Evaluation Criteria:__

- Total system cost
- Detection accuracy
- Non-intrusiveness

Background and Motivation

- Quality inspection is a vital part of any manufacturing process. Occasional mistakes are almost inevitable, especially for companies that produce a wide variety of parts/products, so it is important to catch those mistakes before they reach the customer.
- For small manufacturers, many of the common quality issues are quite simple, such as:
  - The wrong product was dispatched,
  - Too many or too few parts in the box,
  - Not all the required features are present (e.g. there aren't holes where there are meant to be holes).
- Visual inspection is a good method for detecting some of these basic quality issues, but it can be a monotonous task for human inspectors to check each and every part/order which can lead to mistakes slipping through.
- Automating this basic visual inspection can improve overall quality and lets human inspectors focus on the detailed parts of the inspection procedure (such as finish, dimensional tolerances or material composition).
- Participants will develop a low-cost system that uses any domestic low cost camera combined with vision, ML and image processing libraries to deliver an automated visual inspection system able to pass/fail parts or assemblies.

Scope

- At a minimum, the solution would need to:
  - Pass/Fail:
    1. Product **type** at different scales and rotations, e.g. bottles, books, pasta bags, etc.
    2. Considering **presence/absence** detection, e.g. a fixed number of holes, has a label or specific part
    3. Considering **automated measurement**, e.g. distances between holes or product height
    4. Product **composition**, e.g. circuit boards expecting coloured resistors/capacitors in right place
    5. Same as 1 – 3, but observing more than one product simultaneously where possible
  - Implement realistic situations (e.g. same product with different colours, etc.)
- Consider nice to have features and try to implement (e.g. products moving on conveyor belt, barcode reading and traceability, etc.)

Judging Criteria

Teams will be evaluated based on three things:

- Their developed system prototype
- A short 3 minute presentation on their design
- A short (1-2 page or video) tutorial that could be used by an SME to implement their design

The system prototypes and designs will be judged based on three criteria:

- **Total system cost**
- **Detection accuracy**
- **Ease of end-user interaction**
- **Non-intrusiveness** (consider how the solution might work on a real life bolt-on low cost system, e.g. which parts could run on a Raspberry Pi / tablet / desktop/etc., lighting, hardware support, etc.)

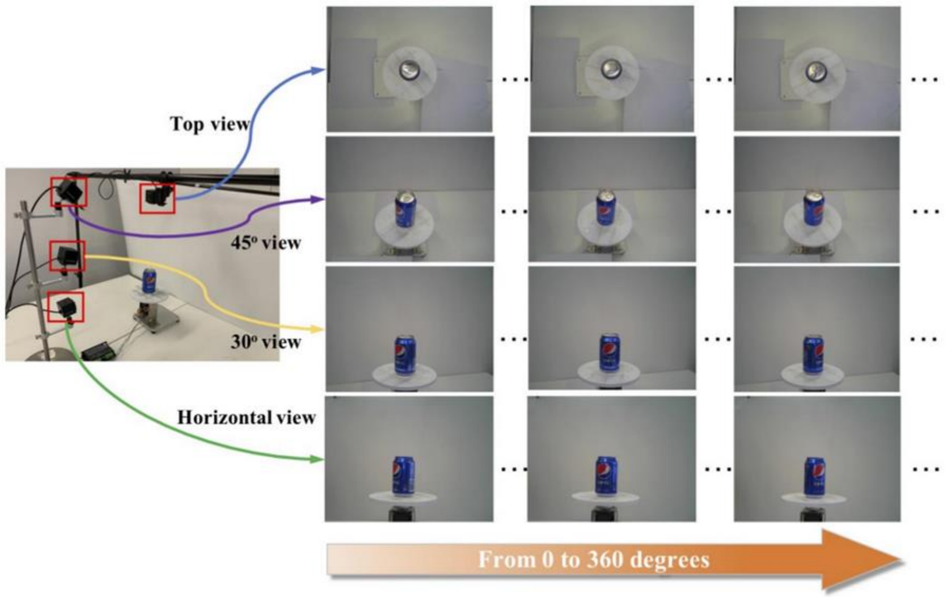
Any additional features or functionality will be considered

Rules

- Teams may think of ideas and approaches as well as do research before the hackathon. However development of the system may not be done beforehand.
- Generate timestamp, part number (or ID) and pass/fail feedback per product in a spread sheet file or other type of format (e.g. CSV, JSON, XML, etc.)
- Failed products must save an associated annotated image (photo) showing what has failed, e.g. square showing a missing label
- The prototype should allow customisation given in a configuration file
- Ideally, all parts of the system should work without the use of proprietary cloud-based services
- You might want to distribute the system parts communicating over Internet, for instance REST APIs for posting products pictures or communicate results
- The solution should be “lightweight“, i.e. able to run on a desktop/laptop or not requiring large computing power

Provided Material

- Consider what you have at home: cans, bottles, egg boxes, any groceries
- We also suggest the 12 Free Retail Image Dataset for Computer Vision, a 15 gb of product images taken from four different views and from 0 to 360 degrees:



This dataset can also help judge what kind and how many photos you will need to train and test your own program:

https://lionbridge.ai/datasets/12-free-retail-image-datasets-for-computer-vision/

Please note that you will need to edit images to make the products fail!

Possible Approaches

Vision

- A mobile phone camera
- Any USB camera, e.g. GoPro-like or sufficiently cheap web camera
- Mobile phone toch or table lamp if lighting is needed

Image processing libraries

- BoofCV https://boofcv.org/ and https://pypi.org/project/PyBoof/
- OpenCV https://opencv.org/
- Zbar http://zbar.sourceforge.net/ and https://pypi.org/project/pyzbar/

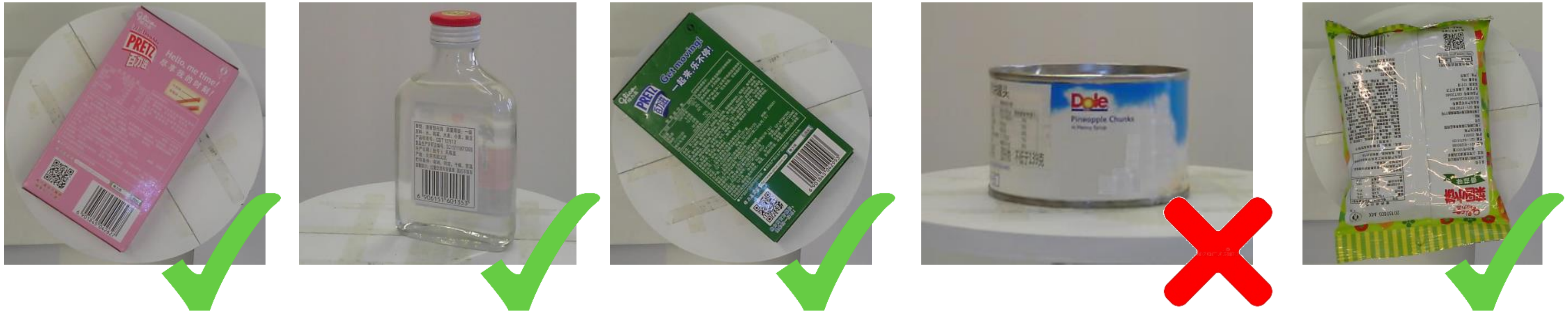
ML libraries

- Tensorflow https://www.tensorflow.org/
- Keras https://keras.io/

Example 1

Do these products have a barcode?

Edited image!



Do all the products of these tea brand have a best before date?

Edited image!

UNIVERSITY OF CAMBRIDGE
Department of Engineering
IfM MANAGEMENT TECHNOLOGY POLICY
UKRI Engineering and Physical Sciences Research Council
University of Nottingham
Institute for Advanced Manufacturing

Example 2

Do these products have same height?

Do these products have a lid?

Edited image!

Example 3

Real life systems:

- **BALUFF Vision Inspection QA** https://www.youtube.com/watch?v=jakSo07JSoc
- **Baumer** https://www.youtube.com/watch?v=KXMPk3FKHgA
- **Compass Automation** https://www.youtube.com/watch?v=_EHoif-ZSqg
- **RNA Vision Inspection & Quality Control** https://www.rnaautomation.com/products/vision-inspection-systems/vision-inspection-quality-control/

#digitalshoestring

24-25 October, 2020

Intelligent Fault Diagnosis Assistance

Challenge:

Develop a smart chatbot system to help operators identify and fix faults

Manufacturing equipment can be very complicated, especially when it comes to fixing faults or errors. In this challenge, participants will develop a chatbot system that helps workers work out what is wrong and guides them through the process of fixing it. There are 3 phases to this challenge:

1) Compile and integrate a library of typical errors/faults with a chatbot platform
2) Provide an user interactive solution
3) Include adaptive diagnostics based on successful instructions

Evaluation Criteria:

> Ease of interaction

> Reliability of assistance

Background and Motivation

- Unplanned downtime can have a substantial impact on SMEs, adaptive diagnostics can help SMEs to quickly identify and fix faults or errors in their manufacturing equipment.
- Learning how to diagnose a machine while attempting to fix it can be a challenge, especially when there are several alternative approaches to deal with the same issue. Furthermore, when technicians leave the company, they take that knowledge with them. An adaptive diagnostics tool can help users to learn how to diagnose equipment quickly, help them to get the machinery up and running again sooner, and improves company retention of the diagnostic knowledge that is gained with each repair.
- Participants will develop a low-cost system that uses an open source chatbot assistant combined with an off-the-shelf adaptive learning library to deliver an interactive, adaptive diagnosis tool that provides the most likely corrective instructions

Scope

- Dealing with faults or errors is typically described in troubleshooting manuals stating the problem, cause (suspected analysis of what is wrong) and solution approaches (guided instructions). However, some solution approaches could work better than others due to a variety of factors such as the equipment model year or operators expertise.
- At a minimum, the solution would need to:
  - Allow operators:
    - Select/input of the closest fault or error
    - Select solution/instructions according to the suspected cause/analysis
    - Rank the selection chosen according to what worked best
  - Implement realistic adaptation features (e.g. add new solutions, links between issues, timing solutions undertaken, etc.)
  - Consider nice to have features and try to implement (e.g. images to help operator identify issues faster, speech interaction, etc.)

Judging Criteria

Teams will be evaluated based on three things:

- Their developed system prototype
- A short 3 minute presentation on their design
- A short (1-2 page or video) tutorial that could be used by an SME to implement their design

The system prototypes and designs will be judged based on three criteria:

- **Total system cost**
- **Adaptability of the diagnosis**
- **Ease of end-user interaction**
- **Non-intrusiveness** (consider how the solution might work on a real life bolt-on low cost system, e.g. which parts could run on a Raspberry Pi / tablet / desktop/etc.)

Any additional features or functionality will be considered

Rules

- Teams may think of ideas and approaches as well as do research before the hackathon. However development of the system may not be done beforehand.
- Operator should interact with a browser interface (e.g. Firefox, Chrome, Zafari, etc.)
- The prototype may have a web-based front-end (e.g Django, Nodejs, Apache, etc.)
- Ideally, all parts of the system should work without the use of proprietary cloud-based services
- The solution should be “lightweight“, i.e. able to run on a desktop/laptop or not requiring large computing power

Provided Material

- Troubleshooting manuals as example (links and files)
- Links to existing chatbot systems (optional to use)
- Links to adaptive learning libraries (optional to use)
- Examples of interactions that will need to be considered for implementation

Possible Approaches

Chatbot systems

- https://botpress.com/
- https://rasa.com/
- https://pypi.org/project/chatbotAI/
- https://github.com/deepmipt/DeepPavlov

Adaptive learning systems (https://www.turing.ac.uk/research/research-projects/adaptive-machine-learning-changing-environments)

- https://gym.openai.com/
  see https://www.learndatasci.com/tutorials/reinforcement-q-learning-scratch-python-openai-gym/
- https://github.com/keras-rl
  see https://www.analyticsvidhya.com/blog/2017/01/introduction-to-reinforcement-learning-implementation/
- https://github.com/sawcordwell/pymdptoolbox
  See https://medium.com/ai%C2%B3-theory-practice-business/reinforcement-learning-part-3-the-markov-decision-process-9f5066e073a2

#digitalshoestring

Example 1

time

Chat Operator and Bot

Op: Hi Chatbot
Bot: How can I help today?
Op: The workpiece milled is not flat
Bot: Ok, are the spindle bearings loose?
Op: No, they are tightened
Bot: Are the gibs of x,y axes loose?
Op: Yes, maybe, they seem loose
Bot: Great, then adjust the gibs' gap. Can you try again and let me know?
Op: It worked well, thank you.

Op: Hello Chatbot
Bot: How can I help today?
Op: This milling isn't flat!
Bot: Ok, are the spindle bearings loose?
Op: Not sure, no.
Bot: Are the gibs of x,y axes loose?
Op: Oh, yes.
Bot: Great, then adjust the gibs' gap. Can you try again and let me know?
Op: It works now

Most faults are related to gibs' gap. Therefore bot updates solutions' priority

Op: Hey Chatbot
Bot: How can I help today?
Op: The workpiece milled isn't flat
Bot: Are the gibs of x,y axes loose?
Op: Indeed
Bot: Great, then adjust the gibs' gap. Can you try again and let me know?
Op: Perfect

Bot rules (high priority first)

1. Check spindle bearings
2. Check gibs x,y axes
3. Check processing amount

1. Check spindle bearings
2. Check gibs x,y axes
3. Check processing amount

1. Check gibs x,y axes
2. Check spindle bearings
3. Check processing amount

Example 2

time

Chat Operator and Bot

Op: Hi Chatbot
Bot: How can I help today?
Op: The workpiece milled is not flat
Bot: check if
1 spindle bearings are loose
2 the gibs x, y axes are loose
3 the processing amount is not too much
Op: It is option 2
Bot: Great, then adjust the gibs' gap. Can you try again and let me know?
Op: It worked well, thank you.

Op: Hello there
Bot: How can I help today?
Op: This milling isn't flat!
Bot: check if
1 spindle bearings are loose
2 the gibs x, y axes are loose
3 the processing amount is not too much
Op: 2 worked well
Bot: Great, then adjust the gibs' gap. Can you try again and let me know?
Op: Satisfied

Most faults are related to gibs' gap. Therefore bot updates solutions' priority

Op: Hey Chatbot
Bot: How can I help today?
Op: The workpiece milled isn't flat
Bot: check if
1 the gibs x, y axes are loose
2 spindle bearings are loose
3 the processing amount is not too much
Op: First one
Bot: Great, then adjust the gibs' gap. Can you try again and let me know?
Op: All done

Bot rules (high priority first)

1. Check spindle bearings
2. Check gibs x,y axes
3. Check processing amount

1. Check spindle bearings
2. Check gibs x,y axes
3. Check processing amount

1. Check gibs x,y axes
2. Check spindle bearings
3. Check processing amount

UNIVERSITY OF CAMBRIDGE Department of Engineering
IfM MANAGEMENT TECHNOLOGY POLICY
UKRI Engineering and Physical Sciences Research Council
University of Nottingham Institute for Advanced Manufacturing

#digitalshoestring

24-25 October, 2020

Virtual Operational Process Planning

paccurate

VOS VOCO

Challenge:

Develop a virtual system to support operational planning that minimises logistics and packing costs

Maximising operational efficiency within an SME is important for minimising delivery time while increasing profit margins. In this challenge, participants will develop an online system for optimising operational logistics and packing. The solution must be able to capture key operational processes of a multi-site company or a shopfloor. There are 3 phases to this challenge:

1) Identify and design a virtual process system
2) Set up and integrate off-the-shelf workflow and packing libraries
3) Simulate and test

__Evaluation Criteria:__

- System integration
- Easy of use
- Broad functionality

Background and Motivation

Operational planning and logistics is an essential activity for small manufacturers. Some example business activities are:

- **Goods in and out** – Products or components need to be moved into and out of the business, as well as between business departments or sites. Tracking deliveries, shipments and stock is essential to avoid waste and loss.
- **Packaging and shipping** – Efficient package planning provides savings in staff time and consignment costs. Moreover, correctly observing package rules (“This way up”, “Do not bend”) reduces shipment damage and subsequent losses.
- **Returns and repair** – Returned goods are to be expected in many businesses. A well organised returns system allows rapid turnaround of responses to customers replacements parts, and fault capture to improve future products.

Background and Motivation

These tasks are so essential that manufacturers always have systems in place. However, many of these systems are manual. Goods-in, goods-out, and fault reporting are often handled with paper forms, whilst packaging is ad-hoc and based on individual employee ability to stack and pack different sized boxes. These systems work for single-product and single-customer relationships, but are strained by an international client base, varied product types and multiple sites which can be typical for even small manufacturers in the twenty-first century.

Example methods for digitisation could use QR codes to identify products or packages, software solvers to generate packing plans for different size and shape boxes, and interactive repair and return forms which can link repair actions and reports to particular products.

Background and Motivation

In this challenge, participants will develop a low-cost system that:

- Can provide work-flow instructions for logistics operations of different complexity; such as goods-in and goods-out, returns and repair, and shipment packaging.
- Integrates with a simulated or model business case by showing the ability to connect product identifiers (QR codes / barcodes / etc) to workflows or instructions and update the information with actions taken, reports, or other information.

The benefit of such a system is two-fold:

- It can provide workers with immediate instructions on what actions to take with a product.
- It can provide high level product, action and shipment tracking across a business.

Scope

- The system must provide plans / instructions for logistics operations within a model business.
- The system should be accessible over a network.
- The system may involve the integration of either or both suggested partner systems, Paccurate or Vos Voco, but ideally integrate both systems.
- Teams will need to decide on the constraints of their system, e.g. focus on packing instructions? Interactive reporting? Provision of instructions?
- The solution will be presented as either a live screen-shared or pre-recorded video. It is acceptable to present some manual steps or workarounds provided it accurately demonstrates how the solution in practice. For example, you may use a laptop/desktop PC to send an HTTP request where your end solution might use a smartphone app. Likewise, you might use a database of test data to simulate the process where a full API integration is not available.

Judging Criteria

Teams will be evaluated based on three things:

- Their developed system prototype
- A short 3 minute presentation on their design
- A short (1-2 page or video) tutorial that could be used by an SME to implement their design

The system prototypes and designs will be judged based on four criteria:

- **Total system cost** (The system should be as affordable as possible)
- **Flexibility** (The system should be able to handle as many different operations as possible)
- **Non-intrusiveness** (The system should not add excessive workload to any of the operations)
- **Reliability** (The system is expected to reliably handle the operations which it encapsulates)

Any additional features or functionality will also be considered

Rules

- Teams may think of ideas and approaches as well as do research before the hackathon. However development of the system may not be done beforehand.
- The system must be accessible in a web browser.
- The solution should either use, or be compatible with, one or both of the provided packages, Vos Voco and Paccurate.

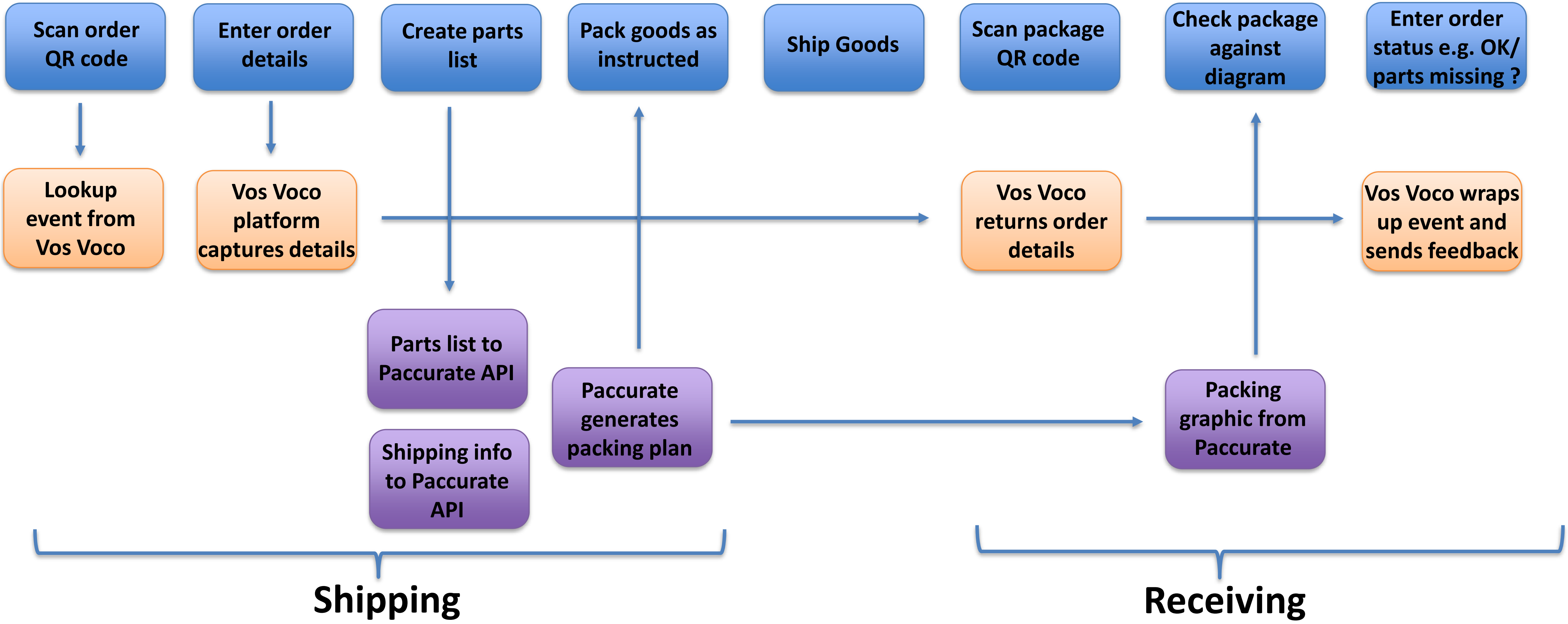
Provided Materials

- Paccurate API
  - Paccurate provides an API for ‘cartonisation’. You can create products of given dimensions, weight, etc, as well as creating rules for them e.g. “this way up”, which are to be packed into containers, again having given dimensions and rules. The output is sets of packaging instructions / diagrams.
- Vos Voco workflow management platform
  - Vos Voco have an online platform for generating workflows, assigning them to companies, and generating QR codes to provide dynamic links to particular workflows and products.

Possible Approaches

- Produce Workflows on the Vos Voco web portal, and generate appropriate QR codes to link to the workflow instructions, in order to capture essential logistics operations, possibly including images, reports or other media.
  - Include the ability to send and receive product repeatedly on a multi-site business
  - Capture changes in the product as it moves through a multi-site business
  - Capture quality information on the product as it moves through a multi-site business
- Produce an interface to the Paccurate API which enables simple product and package creation, with sets of products which need to be shipped together (e.g. kits of products, parts for larger assemblies.)
  - Include an interface for top level tracking of parts in parcels
  - Include an interface for shop floor workers to identify how a product should be packed
  - Produce an ordered list of products to be packed into a box to conform with the packing instructions
- Combine the two systems and demonstrate the ability to capture a whole transaction as illustrated in an example on the next slide.

Example Transaction



#digitalshoestring

DIGITAL MANUFACTURING ON A SHOESTRING

Creating digital solutions that work for manufacturing SMEs

Hackathon

24 – 25 October 2020

digital manufacturing on a shoestring

Pitch-In paccurate ptc VOS VOCO TQC Automation & Test Solutions tqc.co.uk Leak Testing - Automation - Robotics

UNIVERSITY OF CAMBRIDGE Department of Engineering

IfM MANAGEMENT TECHNOLOGY POLICY

UKRI Engineering and Physical Sciences Research Council

University of Nottingham Institute for Advanced Manufacturing