Recognition of inlet wet food in drying process through a deep learning approach

R. Moscetti[1a], S. Massaro[1], D. Monarca[2], M. Cecchini[2], R. Massantini[1b]

[1]Department for Innovation in Biological, Agro-food and Forest system, University of Tuscia, Via S. Camillo de Lellis snc, 01100 Viterbo, Italy; [2]Department of Agriculture and Forest Sciences, University of Tuscia, Via S. Camillo de Lellis snc, 01100 Viterbo, Italy

Abstract

Smart drying is one of the newest and most promising techniques. It is a multi- and inter-disciplinary sector which has potential to guarantee high value end-products by implementing innovative and reliable sensors, resources, tools and practices. Its recent developments embrace various R&D areas, such as computer vision (CV) and deep learning, which deal with allowing computers to understand digital images and videos better than humans. Conventional machine-learning techniques suffer several limitations, mainly due to their inability to process raw data. In fact, in the last few decades, machine learning required considerable domain expertise to mine raw data and extract features from which an algorithm could identify patterns in the input. Deep learning is a novel subfield of machine learning, which embraces methods that allow to discover patterns for detection or classification purposes by using raw data. Consequently, CV in combination with deep learning has the potential to be a powerful Process Analytical Technology tool useful for enhancing the understanding and control of critical process parameters that impact on quality of the final product.

Deep learning was tested for its feasibility as CV tool for the analysis of inlet wet food to drying process. In details, convolutional neural networks (CNNs) were successfully applied for addressing the following tasks: (i) the semantic image segmentation of the inlet product (i.e., recognition between background and product pixels); (ii) the inlet product classification through its segmented image; (iii) the automated selection of optimal settings of drying process parameters.

Results obtained not only represent a step forward in the development of smart dryers able to recognise the inlet wet product, and to set the proper process parameters on its own or as decision support system, but also lay the foundation for further researches on using a computer vision system as PAT tool for smart drying processes.

Keywords: Python; Jupyter; artificial intelligence; machine learning; convolutional neural networks

[a] E-mail: rmoscetti@unitus.it

[b] E-mail: massanti@unitus.it

INTRODUCTION

Among postharvest operations, drying is one of the oldest, typical, effective and viable preservation processes throughout the world. However, it is a relatively complex, dynamic, unsteady and nonlinear process that may suffer from properties of wet material, which may be responsible for low quality end-product (Aghbashlo *et al.* 2015). Consequently, with the aim of circumventing this issue, new drying technologies must be designed around the quality attributes of the wet material.

Among emerging drying technologies, smart drying is one of the newest and most promising techniques. It is a multi- and inter-disciplinary sector which has potential to guarantee high value end-products by implementing innovative and reliable sensors, resources, tools and practices. Its recent development embrace various R&D areas, such as computer vision (CV) together with artificial intelligence, machine learning and deep learning (Moscetti *et al.* 2017), which deal with allowing computers to understand digital images and videos better than humans (i.e. colour, shape and size measurements as well as object segmentation, localization, detection and classification) (Li *et al.* 2015). Consequently, CV in combination with machine learning has the potential to be a powerful Process Analytical Technology (PAT) tool useful for enhancing the understanding and control of critical process parameters that impact on quality of the final product (van den Berg *et al.* 2013).

Conventional machine-learning techniques suffer several limitations, mainly due to their inability to process raw data. In fact, in the last few decades, machine learning required considerable domain expertise to mine raw data and extract features from which an algorithm could identify patterns in the input. Deep learning is a novel subfield of machine learning, which embraces methods that allow discovering patterns for detection or classification purposes by using raw data (LeCun *et al.* 2015).

The objective of the present study was to evaluate the feasibility of using deep learning algorithms for developing smart dryers able to recognise the inlet wet food and to learn how to select the optimal operating conditions (i.e. relative humidity, temperature, air flow rate and duration of process) based on type and characteristics of the inlet wet food.

MATERIAL AND METHODS

Nine species of fruits (i.e. apple, apricot, banana, cherry, kiwifruit, lime, nectarine, pear, plum) and nine species of vegetables (i.e. bell pepper, carrot, champignon mushroom, cherry tomato, cucumber, onion, plum tomato, potato, zucchini) were brought from a local market. Samples were washed and cut into slices of 3-mm thickness, except for apricot, cherry and plum fruits that were longitudinally cut in half. A flat scanner mod. CM2350 (Hewlett-Packard-HP, USA) was used to scan the digital image of the samples. Scanner profiling was performed using a ColorChecker Passport (X-Rite, USA), while image acquisition was carried out with the VueScan PE v9.2.11 software (Hamrick Software, USA). Each image was the average of 3 scans with a resolution equal to 2466×3498×24 (240 dpi and 8 bits per sRGB channel). One hundred samples per batch (i.e. class of product) were acquired. With the aim of circumventing unrealistic results, samples of each batch were randomly split in calibration and prediction subsets (70%) and (30%), respectively.

Image analysis and model development were both performed using interactive Jupyter Notebooks v5.7.4 developed in Python v3.7.2 programming language in combination with various Python packages. Specifically, (i) the conventional image segmentation was carried out with the OpenCV v3.4.5 library, while both (ii) semantic segmentation and (iii) image recognition models were computed using the fast.ai v1.0.51 library running on top of the PyTorch v1.0.1.post2 library. The U-Net fully convolutional network (Ronneberger *et al.* 2015) was used for training the semantic segmentation model (SSM), while a convolutional neural network (i.e. a CNN pre-trained on the ImageNet dataset) was used as base for the development of image classification model (CM). The development of both models was tested on (1) a CPU mod. Ryzen 5 1400 (AMD Inc. CA, USA) and (2) and a GPU mod. RTX 2070 8 GB (NVIDIA Corp. CA, USA). The GPU was used to boost the training step. CPU and GPU performances were compared in terms of time required to train the models. In both cases, a transfer learning approach was applied on a ResNet-34 model (ResNet-34, 2019). The vision.transform module of the fast.ai library was used to perform data augmentation (an image regularization technique) and then to make CNN models invariant to noise, translation, viewpoint, size and illumination of image through small random transformations, which did not change the content of the image itself but affected its pixel values.

The SSM model performance was evaluated through the Intersection over Union (IoU) loss function. IoU consists of the ratio of the number of pixels in common between the target and prediction masks and the total number of pixels present across both masks (eq. 1):

$$(1)\quad \mathrm{IoU} = \frac{\mathrm{target\_pixels} \cap \mathrm{predicted\_pixels}}{\mathrm{target\_pixels} \cup \mathrm{predicted\_pixels}}$$

The CM model was trained using the total error rate as loss function (eq. 2):

$$(2)\quad \text{Total error rate} = \frac{\mathrm{FP+FN}}{\mathrm{P+N}}$$

where,
FP is the Type I error; FN is the Type II error; P and N are the number of real positive and negative cases in the data, respectively.

The optimal learning rates for both SSM and CM models were estimated using the Cyclical Learning Rates approach proposed by Smith (2015), while the optimal number of epochs was chosen at the point when calibration and cross-validation losses began to converge.

RESULTS AND DISCUSSION

Figure 1 shows the 2-step method used to develop both SSM and CM models. Step #1: the SSM model was retrained using masks obtained through conventional segmentation performed using an image thresholding method via the OpenCV library. Conventional segmentation is an operator-dependent and time-consuming process and thus the SSM model was developed to make the process simpler, accurate and automated. Step #2: samples were automatically cropped using the SSM model and then used as input for the development of the classification model. The CM model was retrained by replacing the ImageNet classifier in the last layer with a classifier having two new layers and many targets as the number of classes of the experiment (i.e. 18).

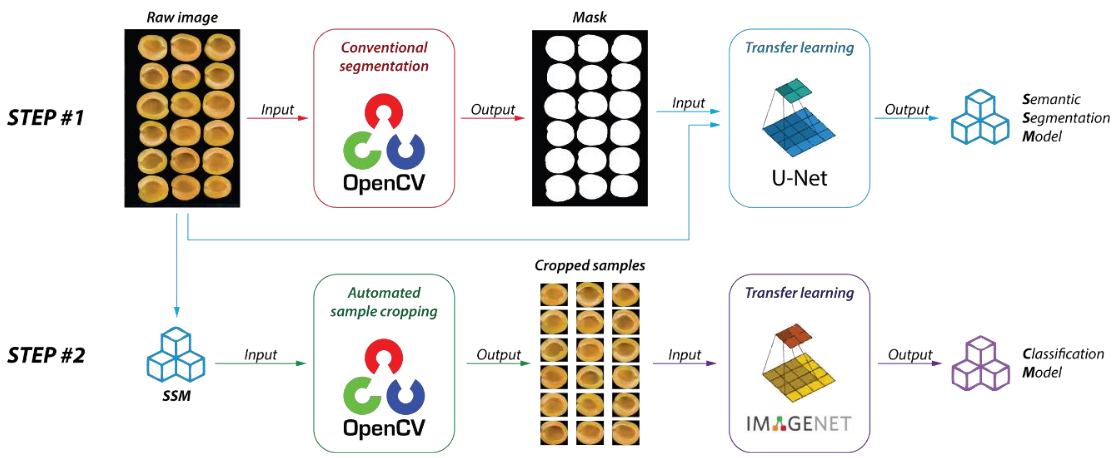

Figure 1. Approach used to develop both SSM (step #1) and CM models (step #2).

The best SSM model was retrained using the U-Net architecture with a learning rate of $5e^{-5}$ across 10 epochs and a batch size equal to 8. It showed excellent prediction performance in terms of Intersection over Union (IoU > 99%) (Figure 2).

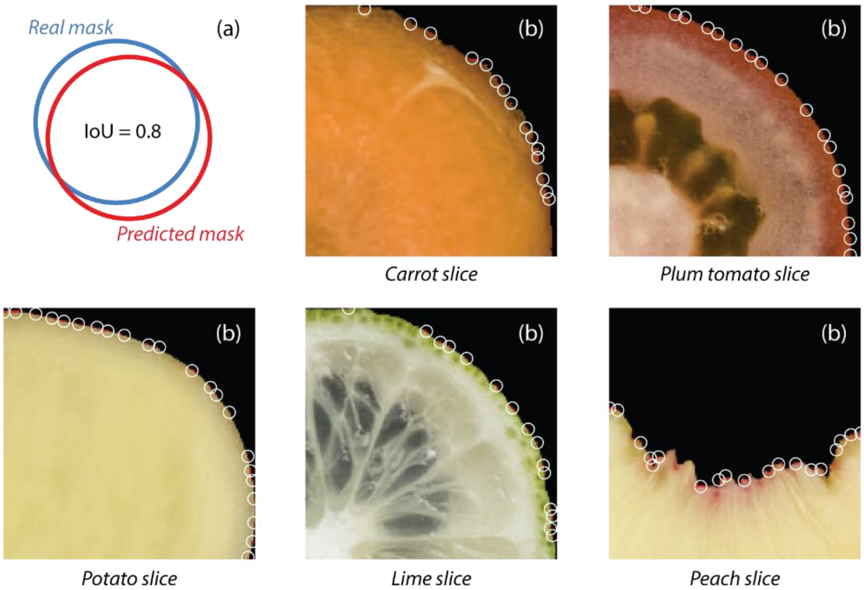

Figure 2. Example of IoU (a) and images of samples with misclassified pixels (b).

The best CM model was obtained with a learning rate of $1e^{-3}$ across 3 epochs and a batch size equal to 64. It showed excellent total error rate in calibration (approx. 0.66%), cross-validation (approx. 0.99%) and prediction (approx. 0.81%). The total error rate was always due to peach slices misclassified as potato or red plum slices (Figure 3). Thus, further studies are needed for fine-tuning the CM model for peach slices recognition.

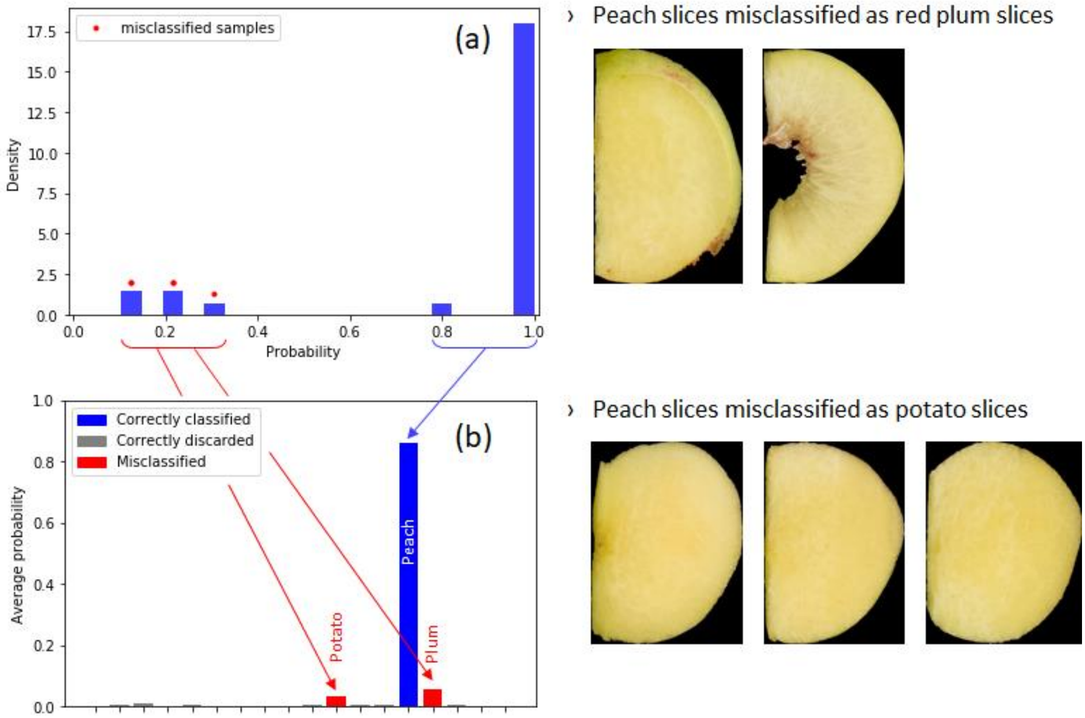

Figure 3. Density histogram of the prediction probability for peach slices (a) and average prediction probabilities computed during peach slices recognition (b).

Table 1 shows some preliminary performance results of commodity hardware based on CPU and GPU. Despite the entry level class of the gaming GPU, its computing performance (i.e. model training time) was extremely encouraging with respect to a multicore CPU. However, further benchmarking tests are ongoing on our computing facility to assess this performance scenario and a detailed progress report will be reported in further studies.

Table 1. Training performance: comparison between CPU and GPU

Model	Learning rate	Epochs	Batch Size	Runtime system	Training time (hh:mm:ss)
CM[a]	$1E^{-03}$	3	64	CPU	00:07:47
				GPU	00:00:26
SSM[b]	$5E^{-05}$	10	8	CPU	09:46:00
				GPU	00:22:31

[a]CM: classification model.
[b]SSM: semantic segmentation model.

CONCLUSIONS

CNNs were used to model both semantic image segmentation and image recognition of inlet wet fruits and vegetables in drying process. The networks produced very good results without any image pre-processing, even though data augmentation was significantly beneficial. Results obtained not only represent a step forward in the development of smart dryers able to recognise the inlet wet product, and to set the proper process parameters on its own or as a decision support system, but also lay the foundation for further researches on using computer vision system, alone or in combination with other sensors, as PAT tool to monitor and control smart drying processes.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge [1] CORE Organic Plus consortium (ERA-NET action) and MIPAAF (Ministero delle politiche agricole alimentari e forestali - Italy) for financial support through the SusOrgPlus project (D.M. 20/12/2017, n. 92350) and [2] the 'Departments of excellence 2018' program (i.e. 'Dipartimenti di eccellenza') of the Italian Ministry of Education, University and Research (MIUR) for the financial support through the 'Landscape 4.0 food, wellbeing and environment' (DIBAF department of University of Tuscia) Moreover, our sincere thanks to Gianpaolo Moscetti and Swathi Sirisha Nallan Chakravatula for the English language revision of the manuscript.

Literature cited

Aghbashlo, M., Hosseinpour, S., Mujumdar, A.S. (2015). Application of Artificial Neural Networks (ANNs) in Drying Technology: A Comprehensive Review, Dry Technol. 33, 1397-1462.

LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep Learning, Nature. 521, 436-444.

Li, J.B., Huang, W.Q., Zhao, C.J. (2015). Machine vision technology for detecting the external defects of fruits - a review. Imaging Sci J. 63, 241-251.

Moscetti, R., Raponi, F., Ferri, S., Conlantoni, A., Monarca, D., Massantini, R. (2017). Real-time monitoring of organic carrot (var. Romance) during hot-air drying using near-infrared spectroscopy, Food Bioprocess Technol. 10, 2046-2059.

ResNet-34 Pre-trained Model for PyTorch. (2019). www.kaggle.com/pytorch/resnet34

Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. arxiv.org/abs/1505.04597

Smith, L.N. (2015). Cyclical Learning Rates for Training Neural Networks. arxiv.org/abs/1506.01186

van den Berg, F., Lyndgaard, C.B., Sørensen, K.M., Engelsen, S.B. (2013). Process analytical technology in the food industry. Trends Food Sci Technol. 31, 27-35.