MACHINE VISION IN MANUFACTURING PROCESSES AND THE DIGITAL TWIN OF MANUFACTURING ARCHITECTURES

Gicu Calin Deac, Crina Narcisa Deac, Cicerone Laurentiu Popa,
Mihalache Ghinea & Costel Emil Cotet

This Publication has to be referred as: Deac, C[rina] N[arcisa]; Deac, C[rina] N[arcisa]; Popa, C[icerone] L[aurentiu]; Ghinea, M[ihalache] & Cotet, C[ostel] E[mil] (2017). Machine Vision in Manufacturing Processes and the Digital Twin of Manufacturing Architectures, Proceedings of the 28th DAAAM International Symposium, pp.0733-0736, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-11-2, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/28th.daaam.proceedings.103

Abstract

This paper describes some results of authors' research in machine vision, as a key technology, essential in connecting the manufacturing processes with the digital twin of the manufacturing architecture. The focus is on designing a system that uses machine vision in combination with deep learning algorithms. The result is a human-like vision system for real-time analysis and interpretation of all video streams of cameras to continuously detect and count the various products on the production line. All detected features and counting information are available in real-time by augmenting on a video screen and streamed to the cloud where they can be easily processed and stored.

Keywords: machine vision; Industry 4.0; augmented reality

1. Introduction

In automated manufacturing, vision systems have been essential in many industries such as: electronics, medical devices, consumer goods, semiconductor, packaging, automotive [1]. Machine visions are implemented mainly for inspecting and identifying parts, measuring dimensions [7], or guiding machines or robots during assembly operations, for pick-and-place [8]. It is not at all clear what the term "computer vision" means. Most people in the field would agree that the use of a single photocell detector to count bottles on a manufacturing line, for example, does not constitute computer vision [2]. They would also agree that the new powerful systems that inspect components and aid robots in assembly most definitely exemplify computer vision. But there appears to be almost a continuum of systems and applications in between.

Research on machine vision has been in progress for over 25 years, yet practical accomplishments have been modest. To date most installed machine vision systems are relatively simple and involve only simple computer processing. In most cases, only silhouette images are utilized and frequently the illumination is specially structured to reduce burden on the pattern recognition computation. Vision is by far the most powerful and versatile sensing mechanism, and once we achieve machine vision with a capability approaching that of human vision at a competitive cost, the installation of machine vision systems will virtually explode [3].

To define our research, we started from the following questions:

1. Is it possible to integrate a distributed vision system across the entire production line?
2. What is the solution for real-time analysis of video streams to identify and count different objects?
3. Is such a solution feasible in terms of deployment cost?

2. Method - Tool

Our research is focused on developing a vision system close to the human vision, using video streams and image processing based on a neural network. The vision system should be able to obtain a real-time classification and to count different products from different stages of processing on an assembly line and send the information to a cloud IIoT application.

After analysing the technical solutions existing on the market at this moment, we designed the following system, whose diagram is presented in Figure 1. The system is composed by several video cameras placed in different points on the production line. The video signal from all cameras is processed by a video multiplexer, resulting a single video stream with all images combined. This video stream is captured by a video capture board and processed by the classification application running on the vision computer. To obtain real-time classification, we used YOLO9000 [4], [5], compiled with CUDA support. The parallel CUDA architecture, gives extra processing power at a low cost. The processing is done in real time (using 4 Nvidia GeForce Titan X video cards, we could analyse the video stream at 60 frames per second). In the classification process the combined image is augmented and tracked.

Fig. 1. Vision system diagram

The video cameras used in this application are colour SD 4:3 aspect ratio with resolution of 720x576 and with composite video output.

The video multiplexer stitches the images provided by the cameras, side by side, top/bottom in a matrix resulting a single video stream with a higher resolution. The video capture card digitises the video output of the video multiplexer and feed the stream to the application.

The application for classification is based on YOLO9000 [4], [5], a convolutional network with anchor boxes which predicts the coordinates of bounding boxes directly using fully connected layers on top of the convolutional feature extractor. YOLO9000 is a real-time framework for detection of more than 9000 object categories by jointly optimizing detection and classification. It uses WordTree (figure 2) to combine data from various sources and a joint optimization technique to train simultaneously on ImageNet and COCO. YOLO9000 is a strong step towards closing the dataset size gap between detection and classification.

The WordTree representation of ImageNet offers a richer, more detailed output space for image classification. Dataset combination using hierarchical classification would be useful in the classification and segmentation domains. Training techniques like multi-scale training could provide benefit across a variety of visual tasks. [5]

Starting with YOLO 9000 pre-trained on ImageNet and COCO we have performed a re-training based on images of different products in different stages of production.

Processing images with YOLO is simple and straightforward. The system resizes the input image to 448 x 448, runs a single convolutional network on the image, and thresholds the resulting detections by the model's confidence (figure 3). A single convolutional network simultaneously predicts multiple bounding boxes and class probabilities for those boxes. YOLO trains on full images and directly optimizes detection performance [5].

Fig. 2. Combining datasets using WordTree hierarchy. [5]

This unified model has several benefits over traditional methods of object detection, being faster and unlike sliding window and region proposal-based techniques, YOLO sees the entire image during training and test time so it implicitly encodes contextual information about classes as well as their appearance [5].

Fig. 3. The YOLO detection system (You Only Look Once) [5]

The detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. The convolutional layers are pretrained on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection (figure 4) [5].

YOLO predicts multiple bounding boxes per grid cell. At training time, we only want one bounding box predictor to be responsible for each object. This leads to specialization between the bounding box predictors. Each predictor gets better at predicting certain sizes, aspect ratios, or classes of object, improving overall recall [6].

YOLO saves the detection and classification information in a text log file. Using a log parser script, we get the latest information and create a Json file containing time stamp in milliseconds, a list of detected objects and their counts. The Json file is uploaded using a secured socket connection to the cloud server, where data is stored in a database and analysed.

Fig. 4. Yolo architecture [5]

Our future work will be focused on testing and implementing the new Nvidia Jetson TK1 developer kit and Vision Works in the same workflow, to cut the prices and create a more simplified and robust architecture. Also, we will extend the system with new functionalities such as filtering damaged items, alarming on work safety incidents (when a person passes in a dangerous zone).

3. Conclusion

By choosing to combine multiple images from different cameras into a single high-resolution stream we have succeeded to integrate a distributed system across the entire production line, that allows the use of a single computer and an application to detect, classify and count all products in different manufacturing stages from the production line. By using YOLO compiled with CUDA support we benefit from fastest detection platform which allows real time classification at a small price.

Limitations: Due to the use of a computer and an application requiring real-time video processing, which results in intense system loading, an efficient cooling system is required to not compromise the stability of the application. A dedicated hardware-based approach would probably be more efficient in terms of long-term exploitation.

4. Acknowledgments

This work was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS/CCCDI – UEFISCDI, project number PN-III-P2-2.1-BG-2016-0437, within PNCIDI III.

5. References

- [1] Abdulrahman, W. (2014). Overview on Industrial Vision Systems, International Journal of Engineering Research & Technology, Vol. 3, Issue 5, pp. 383-386, ISSN: 2278-0181
- [2] Agin G. J. (1980). Computer Vision Systems for Industrial Inspection and Assembly, Computer, Vol. 13, Issue: 5, pp. 11 – 20, ISSN: 0018-9162
- [3] Freeman, H. Machine Vision (1988): Algorithms, Architectures, and Systems, Academic Press, ISBN 0-12-266720-4, USA
- [4] Redmon, J., Farhadi, A. (2016). YOLO9000: Better, Faster, Stronger; Available from: <https://arxiv.org/abs/1612.08242>, Accessed: 2017-08-09
- [5] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection, pp. 779-788, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, ISBN 978-1-4673-8851-1
- [6] R. Girshick, J. Donahue, T. Darrell, and J. Malik. (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. Available from: <https://arxiv.org/abs/1311.2524> Accessed: 2017-08-09
- [7] Hruz, M.; Siroky, J. &Manas, D. (2010). Knoop Hardness Measurement Using Computer Vision, Annals of DAAAM for 2010 & Proceedings of the 21st International DAAAM Symposium, Volume 21, No. 1, ISSN 1726-9679 ISBN 978-3-901509-73-5, Published by DAAAM International, Vienna, Austria, EU, 2010
- [8] Feriancik, M. & Liska, O. (2011). Autonomous Positioning Based on Machine Vision, Annals of DAAAM for 2011, Proceedings of the 22nd International DAAAM Symposium, Volume 22, No. 1, ISSN 1726-9679 ISBN 978-3-901509-83-4, Published by DAAAM International, Vienna, Austria, EU, 2011