A Hardware-Software Co-Design for Object Detection Using High-Level Synthesis Tools

Aiman Badawi, King Abdulaziz University, Jeddah, Saudi Arabia
Muhammad Bilal, King Abdulaziz University, Jeddah, Saudi Arabia

ABSTRACT
Object detection is a vital component of modern video processing systems, and despite the availability of several efficient open-source feature-classifier frameworks and their corresponding implementation schemes, inclusion of this feature as a drop-in module in larger computer vision systems is still considered a daunting task. To this end, this work describes an open-source unified framework which can be used to train, test, and deploy an SVM-based object detector as a hardware-software co-design on FPGA using Simulink high-level synthesis tool. The proposed modular design can be seamlessly integrated within full systems developed using Simulink Computer Vision toolbox for rapid deployment. FPGA synthesis results show that the proposed hardware architecture utilizes fewer logic resources than the contemporary designs for similar operation. Moreover, experimental evidence has been provided to prove the generalization of the framework in efficiently detecting a variety of objects of interest including pedestrians, faces and traffic signs.

KEYWORDS

INTRODUCTION
Recent years have seen the utility of object detection being recognized in modern video processing systems for important tasks such as surveillance, advance driver assistance and urban monitoring etc. Thus, various feature-classifier algorithms and their corresponding implementation schemes on both hardware and software platforms have been proposed and discussed at length by various researchers in the contemporary literature (Cao, Pang, & Li, 2017; Kyrkou, Bouganis, Theocharides, & Polycarpou, 2016; Rettkowski, Boutros, & Göhringer, 2017; Shen, Zuo, Li, Yang, & Ling, 2017; Z. Zhang, He, Cao, & Cao, 2016). Among this variety of approaches, the Convolutional Neural Network (CNN) or Deep Learning (DL) based detectors (Ribeiro, Nascimento, Bernardino, & Carneiro, 2017) have occupied the top place in terms of accuracy in the recent years. However, these top performing techniques require an equivalently high computational cost at the run-time. Moreover, these algorithms have been generally developed without regard for deployment on small embedded systems with limited memory, computational resources and energy sources. The sheer complexity of such systems requires the necessity to engage high-performing dedicated Graphical Processing

DOI: 10.4018/IJECME.2019010105

Copyright © 2019, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
Units (GPU) even on conventional desktop computer. It is due to this reason that despite appearance of a few low-complexity versions (Ren, He, Girshick, & Sun, 2015; L. Zhang, Lin, Liang, & He, 2016), DL-based solutions have yet to see widespread appeal for embedded systems. Furthermore, the dedicated hardware implementation of CNN on FPGA systems requires extensive effort in design and deployment. The problem is exacerbated when newer neural networks need to be incorporated which leads to redesign from scratch. Resultantly, the system-level design of larger computer vision systems demands considerably more time and effort to be spent on the implementation of object detection task alone. Thus, the conventional low-complexity hand-crafted features e.g. Histogram of oriented Gradients (HOG) and Local Binary Pattern (LBP), etc., are still favored by the hardware designers and the embedded system developers (Bilal, 2017; Bilal, Khan, Khan, & Kyung, 2016; Rettkowski et al., 2017) alike. This is the reason that commonly used image processing software libraries such as OpenCV (Bradski, 2000) and Matlab Computer Vision Toolbox (“The Mathworks Simulink”) still support object detection through HOG features and supply corresponding linear Support Vector Machine (SVM) models for classification. The hardware acceleration of these algorithms, however, still requires manual design due to the lack of automated high-level synthesis support for these features. FPGA synthesis tools such as Xilinx Vivado do not include intrinsic modules to support such functionality as well. Although a few research works have employed high-level synthesis tools to develop custom hardware for object detection task, a complete framework for training, testing and hardware deployment of feature-classifier setup is still missing. In this regard, this paper describes a high-level synthesis framework for FPGAs to generate Intellectual Property (IP) core for object detection using low-complexity features in Matlab/Simulink environment. The proposed framework allows the system designers to use object detection feature while developing computer vision systems at the highest abstraction level using common Simulink blocks. The framework can be used to train corresponding SVM models for different objects and realize software-based classification in combination with the developed synthesized hardware as a Hardware-Software (HW-SW) Co-design on an FPGA platform. Thus, all the essential tasks of an object detection task, i.e. training, testing and hardware deployment, can be carried out using a single framework. The source codes for the proposed framework can be downloaded as open-source software\(^1\). These contributions can be summarized as follows:

1. First of its kind open-source Simulink model to generate FPGA IP core for generic object detection;
2. Standard AXI streaming protocol interface for compatibility with generic video processing systems;
3. HW-SW Co-design to test the IP core for real-time object detection;
4. Associated framework for SVM training, software testing and real-time deployment.

The rest of the paper is organized as follows. The next section gives a brief overview of the relevant works reported in the literature. This is followed by the description of the proposed framework. Important results related to the hardware implementation and detection on standard datasets have been provided and discussed later. The paper concludes by highlighting the key points of the conducted work.

**BACKGROUND**

The literature mentions several noteworthy research efforts to come up with low complexity implementation of image features and machine learning classifiers for detection of various objects in software applications and on dedicated hardware platforms such as FPGAs. In this regard utility of SURF and SIFT features has been studied for vehicle model detection (Sperker & Henrich, 2013). HOG features, which are similar to SIFT in computations, were made popular (Dalal & Triggs, 2005) when these showed remarkable detection accuracy on the newly introduced INRIA pedestrian dataset. Since then, almost all subsequent pedestrian detectors have used HOG in some form or the other. In
Learning Patterns of Learners’ Interaction in Learning Management Systems: A Comparative Study of China and Malaysia
Wu Bing, Teoh Ai Ping and Ye Chun Ming (2013). Cases on Assessment and Evaluation in Education (pp. 537-567).
www.igi-global.com/chapter/learning-patterns-learners-interaction-learning/69503?camid=4v1a