Paste-and-Cut: Collective Image Localization and Classification for Real-Time Multi-Camera Object Detection

Young Eun Kang, Woosung Kang, Taehun Lee, and Hoon Sung Chwa
Electronical Engineering and Computer Science
DGIST
Daegu, Republic of Korea
Email: {kye0520, woosungkang, lee_taehun, chwahs}@dgist.ac.kr

***Abstract*—In recent years, object detection has emerged as a crucial task in various real-world applications, including security surveillance, autonomous vehicles, and robotics. However, traditional object detection models face numerous challenges, such as inefficient image processing, inadequate resource utilization, and a failure to consider the different criticality of input images, making it difficult to apply these models for timely inferences in practical applications. To overcome these challenges, this paper proposes a novel object detection framework, called Paste-and-Cut, that utilizes two techniques, image merging (paste) and RoI patching (cut), to optimize resource utilization and improve object detection performance. Additionally, our approach incorporates a dynamic merge size and canvas size decision mechanism to adapt to varying object detection environments. Experimental results obtained from experiments conducted with the MOT dataset demonstrate the effectiveness of our approach in achieving real-time object detection with improved detection accuracy and without generating any deadline miss. As such, Paste-and-Cut provides a promising solution for efficient and accurate real-time object detection in multi-camera scenarios.**

***Index Terms*—Real-time systems, Object detection, Deep neural networks**

I. Introduction

Over the past few years, studies related to Deep Neural Networks (DNNs) have increased by a great amount. This proliferation of DNN research has dramatically accelerated the development of various object detection DNN models, including YOLO [1]. Object detection is considered a vital task in various safety-critical applications such as drones, autonomous driving, surveillance cameras, and much more, which are operated in real-time. Moreover, object detection models are not limited to traditional object detection systems but are also being extended to multi-camera object detection systems. These multi-camera object detection systems provide crucial information about the environment. Timeliness is one of the most significant aspects of these tasks. For instance, in applications such as autonomous vehicles, if the task is not completed within the deadline, the control signal for the brake will be delayed. This delay may cause severe injuries or even fatal accidents. Despite the progress made in DNN models, there still exist challenges in meeting the deadline associated with traditional DNN-based object detection systems. There are three issues to consider with regard to the system model of traditional object detection tasks. Firstly, current systems do not consider an efficient use of available computing resources. Second, traditional object detection models do not consider the issue of prioritizing different areas in an image. Finally, traditional models do not consider dynamically changing deadlines. To overcome the three limitations of current object detection systems, we propose a novel framework, called Paste-and-Cut, that synergetically utilizes Image Merge and Image Patching techniques. Our model is designed to be compatible with any existing DNN-based object detection model. We also propose a Dynamic Decision module to set decisions on merge-image and canvas-image size that directly leads to meeting deadlines for each task. We have implemented the work on PyTorch [2], a popular open-source machine-learning framework developed by Facebook, and evaluated the effectiveness of our approach in terms of accuracy and execution time. Our evaluation results demonstrate that Paste-and-Cut shows no deadline miss at any task and no accuracy loss. Also, we evaluated the accuracy of Paste-and-Cut and showed the execution time breakdown of the whole system model and execution time according to the number of objects per frame to help further understand more about Paste-and-Cut framework.

II. Motivation

Image Merge We exploit the image merge technique to save resource utilization. Image merge refers to pasting multiple images into a single frame. As shown in Figure 1, our experimental results demonstrate that traditional models utilizing a single image only utilize up to 20% of the available resources, while our proposed merged approach utilizing image merging techniques can utilize over 80% of the available resources. This highlights the potential of image merging to enhance resource utilization and address the inefficiencies of traditional object detection models that process one image at a time.

RoI Patching. Regions of Interest (RoIs) in an image may contain critical information for subsequent decision-making processes such as vehicle braking. To enhance the performance of object detection systems, we explore the intuition that important regions may only cover a small proportion of an image, leading to the development of the RoI patching technique. To verify this hypothesis, we conducted an occupancy ratio experiment with the MOT dataset [3], which demonstrated that

979-8-3503-1327-7/23/$31.00 ©2023 IEEE

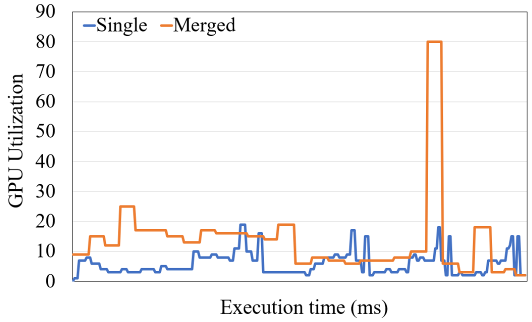

Fig. 1. GPU Utilization Comparison

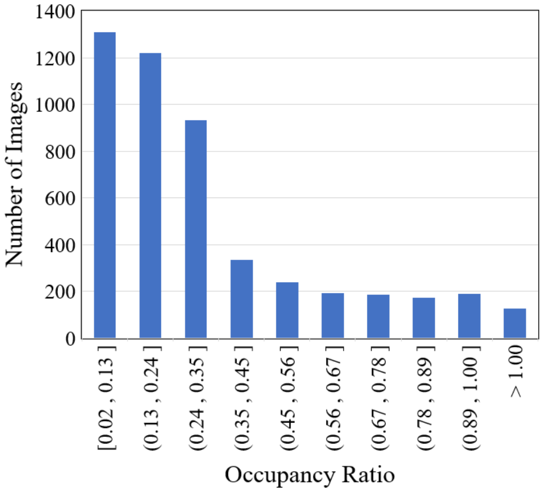

Fig. 2. Occupancy Ratio of RoIs

RoIs occupy only a small portion of the entire image. Figure 2 illustrates the occupancy ratio of the RoIs in relation to their respective images. It indicates that over 77% of the RoIs in the dataset cover less than 50% of the image. These results validate our hypothesis and demonstrate the effectiveness of the RoI patching technique in optimizing object detection performance.

Both image merge and RoI patching techniques face difficulties when applied independently. Merging images with the original object size generates a high-resolution image, which can impact detection accuracy when downsized to meet deadlines. As shown in Figure 3, classification accuracy drops faster with different image widths than localization accuracy. Moreover, the RoI patching technique cannot stand alone since it requires RoI localization to extract the RoIs for patching. Therefore, to overcome these difficulties, we combine the two techniques by utilizing image merge for RoI localization which is relatively insensitive to downsizing and attaching it in front of the patching module.

III. System Design

Our proposed framework, Paste-and-Cut, aims to enable real-time multi-camera object detection tasks based on deep neural networks. The main approach of our framework involves utilizing a combination of image merging and RoI

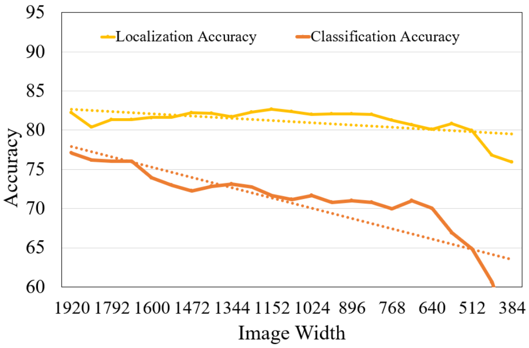

Fig. 3. Image Merge Accuracy Results on Downsizing

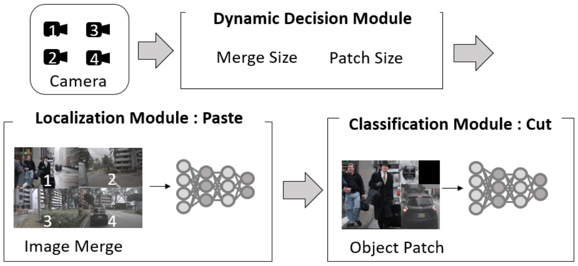

Fig. 4. Overview of system design

patching schemes, along with dynamic resizing strategies to accommodate varying timing constraints. The system architecture is comprised of a Dynamic Decision Module, a Localization Module: Paste, and a Classification Module: Cut, as depicted in Figure 4.

A. Dynamic Decision Module

To dynamically meet the required deadline, our system incorporates a merge size and patch size decision module. To determine the optimal combination, we use a list of merge and patch size combinations based on detection accuracy, which was derived from prior experiments. The list is sorted in descending order of accuracy according to profiled accuracy of merge-patch combinations. We also take into consideration the frame density when selecting the combinations. We define frame density as the number of objects in a frame, as real-time object detection tasks exhibit high similarity between consecutive frames. When the previous frame contains a small number of objects, it is highly likely that the next frame will also contain a small number of objects.

B. Localization Module: Paste

The localization module is responsible for extracting RoIs from the input image. In order to effectively localize the RoIs, we merge the images into a single image that matches the size determined by the decision module.

C. Classification Module: Cut

This classification module is responsible for patching the detected RoIs to a single canvas. To achieve this, two factors need to be considered: 1) RoI size and 2) patch algorithm. The

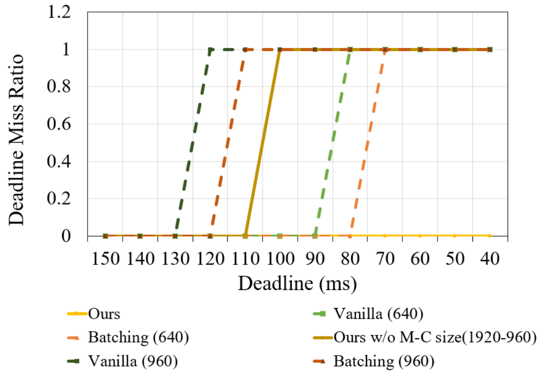

Fig. 5. Deadline Miss Ratio

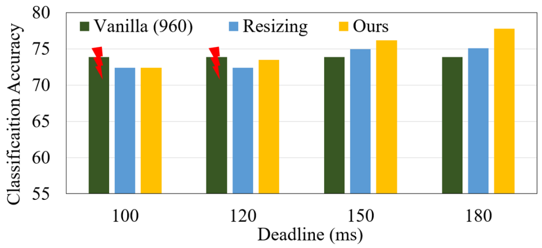

Fig. 6. Accuracy Comparison

size of the detected object to be patched into the canvas is a critical factor that significantly impacts the overall accuracy of the system model. In order to determine the optimal object size options, we conducted an experiment that analyzed the relationship between object size and accuracy changes based on different image sizes. We have selected the image size based on the experiment results.

In order to patch RoIs to a single canvas frame, we employ a recursive packing algorithm. This algorithm involves the placement of rectangle objects on a frame, which is a known NP-hard problem. To resolve this issue we adopt the Priority Heuristic Recursive Packing Algorithm, which is a simple heuristic approach that minimizes canvas waste by selecting the optimal method for patching RoIs in a frame [4].

IV. Evaluation

A. Experiment Setup

Our experiments were conducted on a desktop machine running Ubuntu 18.04.4 and CUDA 11.3, PyTorch 1.12.0. It is equipped with Intel® Core™ I7-8700 CPU @ 3.2 GHz, GeForce GTX 1050 GPU, and 16 GB memory. We evaluated our model using the Multi-Object Tracking (MOT) dataset. Our DNN-based object detection model is based on YOLOv5.

B. Deadline Miss Ratio

We show the main benefit of exploiting our model in a real-time object detection system in terms of deadline. The deadline miss ratios for various models were analyzed and presented in Figure 5. The baseline model with 640 and 960 image sizes exhibited a high deadline miss ratio. Although the batching model performed better than the vanilla model, it still showed deadline misses starting from 70ms and 110ms for 640 and 960 image sizes, respectively. In contrast, our proposed model with a dynamic image size decision module demonstrated no deadline misses at any deadline, indicating its suitability for real-time object detection tasks with dynamically changing deadlines.

C. Detection Accuracy

The comparison of average classification accuracy with the baseline model of image size 960 and resizing method is illustrated in Figure 6. The results demonstrate that as the deadline increases our proposed method outperforms the baseline and resized model. The increase in detection accuracy for the image resizing method decreases as the deadline increases, whereas our proposed method continues to show improvement. This can be attributed to the saturation of detection accuracy as the image size increases, and our proposed method's ability to handle object size similar to the original image size as the deadline increases.

V. Conclusion

In conclusion, this paper presents a novel object detection framework designed specifically for real-time multi-camera scenarios. To address these challenges, we propose an approach that combines two techniques: image merging and RoI patching. We also introduce a dynamic merge size and canvas size decision mechanism that can adapt to varying object detection environments. Our experimental results on the MOT dataset demonstrate the effectiveness of our approach in achieving real-time object detection in terms of deadline miss and detection accuracy. The application of the dynamic scaling method to the system model has enabled our approach to meet deadline constraints in real-time applications. Overall, this paper proposes an efficient object detection framework that addresses the challenges of traditional models in multi-camera scenarios, making it suitable for real-world applications.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant (NRF-RS-2023-00213309), the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant (IITP-2022-0-01053), and the Electronics and Telecommunications Research Institute (ETRI) grant (23ZD1160), funded by the Korea government (MSIT).

References

[1] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. ”You only look once: Unified, real-time object detection.” In Proceedings of the IEEE conference on computer vision and pattern recognition 2016. p. 779-788.

[2] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., and Chintala, S. “Pytorch: An imperative style, high-performance deep learning library”, Advances in neural information processing systems, 2019, 32.

[3] Milan, A., Leal-Taixé, L., Reid, I., Roth, S., and Schindler, K. ”MOT16: A benchmark for multi-object tracking.” arXiv preprint arXiv:1603.00831, 2016

[4] Zhang, D., Shi, L., Leung, S. C., and Wu, T. ”A priority heuristic for the guillotine rectangular packing problem.” Information Processing Letters 116.1 (2016): 15-21.