PerConfigure: Periodically Configure the Video Pipeline Based on a Reinforcement Learning approach

Written by Zhaoliang He[1], Chen Tang[1], Zhi Wang[2]
[1]Department of Computer Science and Techonology, Tsinghua University
[2]Tsinghua Shenzhen International Graduate School, Tsinghua University

Abstract

AAAI creates proceedings, working notes, and technical reports directly from electronic source furnished by the authors. To ensure that all papers in the publication have a uniform appearance, authors must adhere to the following instructions.

Congratulations on having a paper selected for inclusion in an AAAI Press proceedings or technical report! This document details the requirements necessary to get your accepted paper published using PDFLaTeX. If you are using Microsoft Word, instructions are provided in a different document. AAAI Press does not support any other formatting software.

- You must use the 2020 AAAI Press LaTeX style file and the aaai.bst bibliography style file, which are located in the 2020 AAAI Author Kit (aaai20.sty and aaai.bst).
- You must complete, sign, and return by the deadline the AAAI copyright form (unless directed by AAAI Press to use the AAAI Distribution License instead).
- You must read and format your paper source and PDF according to the formatting instructions for authors.
- You must submit your electronic files and abstract using our electronic submission form **on time.**
- You must pay any required page or formatting charges to AAAI Press so that they are received by the deadline.
- You must check your paper before submitting it, ensuring that it compiles without error, and complies with the guidelines found in the AAAI Author Kit.

Introduction

With the increasing demand for continuous video analysis in public safety and transportation, more and more cameras are being deployed to various locations. Video analysis can be completed according to the video analysis application built by different models, which can free the staff from complex and boring tasks or search through massive amounts of video data to find what you're looking for quickly. In recent years, we have also witnessed the emergence of a large number of excellent models for target detection.

For the collected video, the classical computer vision and deep neural network technology are generally used for video analysis. A video analysis application consists of a pipeline of several video processing modules, typically including a decoder, a selective sampling frame application, and a target detector. Such a pipeline always having multiple ”knobs”, such as frame rate, resolution, and model (for example, MobileNet, ResNet, or InceptionResNet), for different combinations of values of knobs are called different configurations. Since video analysis is a very complex process, we pay much attention to the consumption of resources in the calculation process, and the accuracy of inference is also our focus. Therefore, the problem that follows is how to balance the consumption of resources and accuracy. Choosing different configurations will affect the resource consumption and accuracy caused by video analysis. For example, using a complex model and high resolution can obviously accurately detect the target object, but it also requires more computing resources. However, choosing a simple model and low resolution can significantly reduce resource consumption, although it reduces the accuracy to some extent. And in the case of a highway video analysis, due to the rate of car travel cannot be predicted in advance, so when the car driving slowly(or static) because of the traffic jam, we can choose a lower frame rate (such as 1 FPS) without having to use a fixed on the whole video higher frame rate, this can significantly reduce resource consumption, but does not affect the accuracy of the video analysis. The best configuration for a video analytics pipeline alsovaries over time, often at a timescale of minutes or even seconds(Jiang et al. 2018), so our goal is to find a range of ”best fit” configurations that takes up the minimum amount of computing resources and is accurate to the desired threshold. The most intuitive way to solve this problem is to find the best solution by exhaustive all configurations, but the number of possible configurations grows exponentially, and thousands of configurations can be combined with just a few knobs, so exhaustive configuration is a highly unrealistic approach.

In this paper, we propose an approach based on reinforcement learning, which can skillfully use the temporal and spatial characteristics of video to select the "most appropriate" configuration, thus solving a difficult optimal configuration decision problem (reduced to linear time complexity) in a very low-cost way:

- Time related
- Spatial correlation
- Independent configuration

Related Works

Acknowledgments

AAAI is especially grateful to Peter Patel Schneiderfor his work in implementing the aaai.sty file, liberally using the ideas of other style hackers, including Barbara Beeton. We also acknowledge with thanks the work of George Ferguson for his guide to using the style and BibTeX files — which has been incorporated into this document — and Hans Guesgen, who provided several timely modifications, as well as the many others who have, from time to time, sent in suggestions on improvements to the AAAI style.

The preparation of the LaTeX and BibTeX files that implement these instructions was supported by Schlumberger Palo Alto Research, AT&T Bell Laboratories, Morgan Kaufmann Publishers, The Live Oak Press, LLC, and AAAI Press. Bibliography style changes were added by Sunil Issar. \pubnote was added by J. Scott Penberthy. George Ferguson added support for printing the AAAI copyright slug. Additional changes to aaai.sty and aaai.bst have been made by the AAAI staff.

Thank you for reading these instructions carefully. We look forward to receiving your electronic files!

References

Jiang, J.; Ananthanarayanan, G.; Bodik, P.; Sen, S.; and Stoica, I. 2018. Chameleon: scalable adaptation of video analytics. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, 253–266.