PARALLELIZATION OF SEQUENTIAL COMPUTER VISION ALGORITHMS ON BIG-DATA USING DISTRIBUTED CHUNK-BASED FRAMEWORK

By

Norhan Magdi Sayed Osman

A Thesis Submitted to the
Faculty of Engineering at Cairo University
in Partial Fulfillment of the
Requirements for the Degree of
MASTER OF SCIENCE
in
Electronics and Communications Engineering

FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT
2018

PARALLELIZATION OF SEQUENTIAL COMPUTER VISION ALGORITHMS ON BIG-DATA USING DISTRIBUTED CHUNK-BASED FRAMEWORK

By

Norhan Magdi Sayed Osman

A Thesis Submitted to the
Faculty of Engineering at Cairo University
in Partial Fulfillment of the
Requirements for the Degree of
MASTER OF SCIENCE
in
Electronics and Communications Engineering

Under the Supervision of

Prof. Dr. Hossam Aly Hassan Fahmy
Professor
Electronics and Communications Engineering Department
Faculty of Engineering, Cairo University

Dr. Mohamed Mohamed Rehan
Chief Technology Officer
Avidbeam Technologies

FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT
2018

PARALLELIZATION OF SEQUENTIAL COMPUTER VISION ALGORITHMS ON BIG-DATA USING DISTRIBUTED CHUNK-BASED FRAMEWORK

By

Norhan Magdi Sayed Osman

A Thesis Submitted to the
Faculty of Engineering at Cairo University
in Partial Fulfillment of the
Requirements for the Degree of
MASTER OF SCIENCE
in
Electronics and Communications Engineering

Approved by the Examining Committee:

Prof. Dr. Hossam Aly Hassan Fahmy, Thesis Main Advisor

Dr. Mohamed Mohamed Rehan, Advisor
Chief Technical Officer, AvidBeam Technologies

Prof. Dr. Elsayed Eissa Abdo Hamyed, Internal Examiner

Prof. Dr. Khaled Mostafa Elsayed, External Examiner
Faculty of Computers and Information, Cairo University

FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT
2018

Engineer's Name: Norhan Magdi Sayed Osman
Date of Birth: 27/5/1990
Nationality: Egyptian
E-mail: nmbuckla@live.com
Phone: 01224209955
Address: 12D, Maadi Gardens Compound, Daary Maadi, Katamya

Registration Date: 1/3/2014
Awarding Date: / /2018
Degree: Master of Science
Department: Electronics and Communications Engineering

Supervisors:

Prof. Dr. Hossam Aly Hassan Fahmy
Dr. Mohamed Mohamed Rehan

Examiners:

Prof. Dr. Hossam Aly Hassan Fahmy	(Thesis Main advisor)
Dr. Mohamed Mohamed Rehan	(Advisor)
Chief Technology Officer, AvidBeam	
Prof. Dr. Elsayed Eissa Abdo Hamyed	(Internal examiner)
Prof. Dr. Khaled Mostafa Elsayed	(External examiner)
Faculty of Computers and Information, Cairo University	

Title of Thesis:

PARALLELIZATION OF SEQUENTIAL COMPUTER VISION ALGORITHMS ON BIG-DATA USING DISTRIBUTED CHUNK-BASED FRAMEWORK

Key Words:

BiG-Data; Computer Vision; Parallel Computing; Video Processing; Video-Chunks

Summary:
In this thesis, we propose a complete framework that enables big-data frameworks to run sequential computer vision algorithms in a scalable and parallel way. Our idea is to divide the input video files into small chunks that can be processed in parallel without affecting the quality of the resulting output. We developed an intelligent data grouping that enables chunk-based framework to distribute these data chunks among the available resources and gather the results out of each chunk faster than the standalone sequential application.

Acknowledgements

I would like to thank my father, my husband and the rest of my family for being patient and supportive during my MSc. period. They believed in me and encouraged me to continue my work with endless love and understanding. I would also like to thank my supervisors, Dr. Hossam Fahmy and Dr. Mohamed Rehan for their continuous support, patience and guidance to me in order to enhance my work and correct my mistakes during this thesis. I really want to thank all my colleagues at AvidBeam who stood beside me to understand several computer vision basics and helped me to mature my developed work. I would like to thank Dina Helal, Menna Ghoneim, Rana Morsi, Ahmed Hegazy, Eslam Ahmed, Mickael Sedrak, Raghda Hassan and Afnan Hassan as well. Special thanks goes to Dr. Hani El Gebaly, Hossam Sami and AvidBeam for providing support, time and hardware capabilities to develop and test my proposed work and reach the final results.

Dedication

To the soul of my *mother*, I know your are proud of me.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AM	Application Master
API	Application programming interface
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CV	Computer Vision
DAG	Direct Acyclic Graph
DCB-Framework	Distributed Chunk-Based Framework
DBMS	Database Management System
DSFU	Data Splitting and Feed Unit
DMRMU	Decision Making and Resources Mapping Unit
FIFO	First In First Out
FPS	Frames Per Second
GFS	Googles File System
GPU	Graphyical Prcoessing Unit
GOP	Group of pictures
HDFS	Hadoop Distributed File system
JAR	Java Archive
JVM	Java Virtual Machine
LPR	License Plate Recognition
MPEG	Moving Picture Experts Group
NM	Node Manager
NFS	Network File Systems
OCR	Optical Character Recognition
OpenCV	Open Source Computer Vision Library
OpenMP	Open Multi-Processing
OS	Operating System
PC	Personal Computer
QoS	Quality of Service
RCU	Resources Calculation Unit

RDD	Resilient Distributed Dataset
ROI	Region Of Interest
RTSP	Real Time Streaming Protocol
SIMR	Spark In MapReduce
SQL	Structured Query Language
RM	Resource Manager
UI	User Interface
VM	Virtual Machine
VP	Video Processing
YARN	Yet Another Resource Negotiator

List of Terms

Bolts	Data processing entities at Storm.
Executor	A thread runs at Storm worker process.
Kafka	An open source stream processing tool.
Kestrel	A simple, distributed message queue system
MapReduce	Parallel data processing scheme with two stages: Map and Reduce
Nimbus	A Java daemon runs at Storm master node.
Scheduler	An algorithm used to distribute processing resources on big-data tools
Storm	An open source big-data tool
Spouts	Storm data sources.
Supervisor	A Java daemon runs at Storm worker node
Thrift	A software framework used for scalable cross-language services development
Topology	A directed graph that connects Storm spouts and bolts together.
Zookeeper	An entity that governs the communication between Storm nimbus and supervisors

Abstract

The research presented in this thesis addresses the parallelization of sequential computer vision algorithms, such as motion detection, tracking, etc., on big-data tools. Computer vision sequential algorithms have restrictions on how the video frames should be processed. In these algorithms, the inter-relation between successive frames is important part of the algorithm sequence as the result of processing one video frame depends on the result of the previous processed frame(s).

Most of the present big-data processing frameworks distribute the input data randomly across the available processing units to utilize them efficiently and preserve working load fairness. Therefore, the current big-data frameworks are not suitable for processing video data with inter-frame dependency. When processing these sequential algorithms on big-data tools, splitting the video frames and distributing them on the available cores will not yield the correct output. Consequently, the advantage of the processing sequential algorithms on big-data framework becomes limited only to certain cases where video streams are coming from different input sources.

In this thesis, we propose a complete framework that enables big-data tools to execute sequential computer vision algorithms in a scalable and parallel way with limited modifications. Our main objective is to parallelize the processing operation in order to speed up the required processing time. The main idea is to divide the input big-data video files into small chunks that can be processed in parallel without affecting the quality of the resulting output. We have developed an intelligent data grouping algorithm that distributes these data chunks among the available processing resources and gather the results out of each chunk. A parallelized chunk-based data splitter was used to provide the input data chunks concurrently for parallel processing. Then, our grouping algorithm makes sure that all frames that belong to the same chunk are distributed in order to the associated processing cores.

To evaluate the performance of the developed chunk-based framework, we conducted several experimental tests. We used Apache Storm as our big-data framework for its real-time performance. Storm framework was modified to support input video frame splitting and parallel execution. We examined the behavior of our proposed framework against different number of chunks over one hour testing videos. In our evaluation, we used several sequential computer vision algorithms including face detection, video summarization, license plate recognition (LPR), and heatmaps. Those algorithms were integrated into