

ການກວດຈັບຄວາມໄວລົດ ແລະ ການນັບຈຳນວນລົດ ໂດຍໃຊ້ CNN-YOLO

Vehicles Counting and Speed Detection Using CNN-YOLO

ລັດສະໝີ ຈິດຕະວົງ, ດາຊິງ ໂຊ່ງຢັງເຊັງ, ໄຊຕະພາບ ຮວງ, ຫວີເລົ່າ ທອງພັນ, ບຸນເລີດ ວົງມະນີ
l.chidtavong@nuol.edu.la, daxionginfo@gmail.com, Xaytaphabpsl@gmail.com,
Vuelaonthongphanh@gmail.com, bounleuthvmn@gmail.com

ຜາກວິຊາວິທະຍາສາດຄອມພິວເຕີ, ຄະນະວິທະຍາສາດທຳມະຊາດ, ມະຫາວິທະຍາໄລແຫ່ງຊາດ

ບົດຄັດຫຍໍ້

ງານວິໄຈນີ້ມີຈຸດປະສົງເພື່ອພັດທະນາແມ່ແບບ (Model) ຂອງປະເພດຍານພາຫະນະ ແລະ ພັດທະນາລະບົບການກວດຈັບເພື່ອຈະໄປສອນໃຫ້ລະບົບຮັບຮູ້ການກວດຈັບຄວາມໄວ ແລະ ນັບຈຳນວນລົດຈາກພາບວິດີໂອ ຫຼື ກ້ອງຖ່າຍພາບວິດີໂອຕາມທ້ອງຖະໜົນທີ່ມີກ້ອງວົງຈອນປິດ, ໂດຍມີເປົ້າໝາຍເພື່ອຊ່ວຍການກວດຈັບຄວາມໄວ ແລະ ນັບຈຳນວນລົດ ຕາມແຕ່ລະສະຖານທີ່ທີ່ມີກ້ອງຕິດຕັ້ງຢູ່. ການພັດທະນາແມ່ແບບ (Model) ແລະ ພັດທະນາລະບົບການກວດຈັບແມ່ນນຳໃຊ້ເຕັກນິກຫຼັກການຂອງ CNN – YOLO Algorithm ເພື່ອຄຳນວນຫາຄຸນລັກສະນະຂອງຮູບພາບ.

ຜ່ານການພັດທະນາ ແລະ ທົດລອງລະບົບເຫັນວ່າມີຄວາມຖືກຕ້ອງ ໂດຍສະເລ່ຍປະມານ 90% ຕາມແຕ່ລະ Frame ວິດີໂອທີ່ສົ່ງເຂົ້າຜ່ານລະບົບກວດສອບ ເຊິ່ງຊຸດຂໍ້ມູນທີ່ໃຊ້ໃນການວິໄຈພັດທະນາ ແລະ ທົດສອບລະບົບ Model ເປັນຮູບພາບຖ່າຍຈາກກ້ອງຖ່າຍຮູບພາບທົ່ວໄປ ໂດຍຖ່າຍຕາມແຕ່ລະປະເພດລົດເຊັ່ນ: ລົດຈັກຈຳນວນ 1.000 ຮູບຂຶ້ນໄປ, ລົດໃຫຍ່ຈຳນວນ 1.000 ຮູບຂຶ້ນໄປ ແລະ ລົດຖີບຈຳນວນ 1.000 ຮູບຂຶ້ນໄປ ເປັນຊຸດຂໍ້ມູນສຳລັບການຝຶກຝົນໃຫ້ລະບົບຮັບຮູ້, ສຳລັບການວິໄຈຄົ້ນຄວ້າຄັ້ງນີ້ຜົນການທົດລອງຂອງລະບົບກວດຈັບຄວາມໄວ, ນັບຈຳນວນລົດ ມີຄວາມສາມາດໃນການປະມວນຜົນໃຊ້ເວລາປະມານ 0,2 ວິນາຕໍ່ Frame, ເຊິ່ງສາມາດກວດຈັບຄວາມໄວ, ນັບຈຳນວນລົດໄດ້ສະເພາະລົດທີ່ຫັນໜ້າເຂົ້າຫາໜ້າກ້ອງເທົ່ານັ້ນ ເຊິ່ງຜົນການທົດລອງເຫັນວ່າສາມາດກວດຈັບຄວາມໄວ ແລະ ນັບຈຳນວນລົດໄດ້ໂດຍສະເລ່ຍປະມານ 90% ຈາກວິດີໂອກ້ອງວົງຈອນປິດ.

Abstract

The purpose of this research paper is to develop a model for vehicles detection and to develop a learning system to be able to compute a speed of vehicles, and counting number of vehicles from a video file or from a video camera on streets by CCTV. The goal is to help Lao traffic system have a tool for detecting speed and counting numbers of vehicles especially on main roads. The model and the system are developed by using CNN-YOLO algorithm to detect image and video features.

The development consists of 2 main steps, the system training and the system testing. A result of the research show that the system's accuracy for dataset checking on each frame video is on average 90%. The dataset is from camera snapshots, generally they are each type of vehicles such as motorcycle 1,000 images minimum, car 1,000 images minimum and bike 1,000 images minimum. The processing time is 0.2 seconds per frame and the system can detect and count only the vehicles that moving to the camera.

Keywords: CNN, YOLO, Speed Detection, Vehicle Recognition

1. ພາກສະເໜີ

ໃນປັດຈຸບັນເຕັກໂນໂລຊີໄດ້ມີບົດບາດຫຼາຍຂຶ້ນເນື່ອງຈາກຄວາມຕ້ອງການຂອງມະນຸດເຮົາທີ່ຕ້ອງການຄວາມສະດວກສະບາຍ ແລະ ປະສິດທິພາບຄວາມຖືກຕ້ອງໃນການເຮັດວຽກຫຼາຍຂຶ້ນ ໂດຍການພັດທະນາດ້ານເຕັກໂນໂລຊີໃນດ້ານຕ່າງໆອອກມາຊ່ວຍແກ້ໄຂບັນຫາເຊັ່ນ: ລະບົບເຄືອຂ່າຍທີ່ເອົາມາໃຊ້ເປັນສື່ກາງໃນການຈັດການຂໍ້ມູນ, ການຕິດຕໍ່ສື່ສານ, ກວດສອບຂໍ້ມູນ ແລະ ຮັກສາຄວາມປອດໄພຕ່າງໆຂອງອົງກອນ, ກະຊວງ ແລະ ບໍລິສັດຕ່າງໆ. ເຊິ່ງໃນຫຼາຍບົດທີ່ຜ່ານມາການຄົ້ນຄວ້າການກວດຈັບວັດຖຸເປັນຫົວຂໍ້ທີ່ໄດ້ຮັບຄວາມສົນໃຈ ແລະ ເປັນທີ່ນິຍົມຫຼາຍໃນທາງດ້ານການສຶກສາ, ຄົ້ນຄວ້າ, ວິໄຈ ແລະ ການພັດທະນາເຕັກນິກວິທີການ ເພື່ອນຳໄປໃຊ້ວຽກຕົວຈິງ. ເນື່ອງຈາກໃນປັດຈຸບັນອົງກອນ ຫຼື ກະຊວງຕ່າງໆໄດ້ໃຫ້ຄວາມສຳຄັນກ່ຽວກັບການຈັດການລະບົບຮັກສາຄວາມປອດໄພໃນການຂັບຂີ່ລົດໃນທ້ອງຖະໜົນໃຫ້ມີຄວາມປອດໄພ ແລະ ນັບຈຳນວນລົດການເຂົ້າອອກຂອງແຕ່ລະເສັ້ນທາງ, ໃນນັ້ນມີນັກຄົ້ນຄວ້າວິໄຈສຶກສາຄົ້ນຄວ້າທີ່ກ່ຽວຂ້ອງກັບການພັດທະນາແມ່ແບບ ແລະ ລະບົບການກວດຈັບວັດຖຸ, “Detection of Scratches on Cars by Means of CNN and R-CNN” ເປັນບົດຄົ້ນຄວ້າວິໄຈໜຶ່ງທີ່ຄົ້ນຄວ້າໂດຍ (Cesar G. Pachón-Suescún, Javier O. Pinzón-Arenas, Robinson Jiménez-Moreno, 2019) ເປັນການພັດທະນາລະບົບແມ່ແບບການກວດຈັບລົດ ແລະ ຫາຮອຍຂີດຂ່ວນໃນລົດໂດຍໃຊ້ຫຼັກການ CNN ແລະ R-CNN. ນອກຈາກນີ້ຍັງມີບົດຄົ້ນຄວ້າວິໄຈ “Traffic Signs Detection System by Using Deep Learning” ເປັນບົດຄົ້ນຄວ້າວິໄຈໜຶ່ງທີ່ຄົ້ນຄວ້າໂດຍ (Natthapat.S, 2019) ເປັນການຄົ້ນຄວ້າວິໄຈການກວດຈັບ ແລະ ຈຳແນກປ້າຍຈາລະຈອນຕາມເສັ້ນທາງໂດຍການນຳໃນຫຼັກການວິທີ Deep Learning.

ບັນຫາອຸປະຕິເຫດຕາມທ້ອງຖະໜົນນັບເປັນບັນຫາສັງຄົມທີ່ສຳຄັນທີ່ສຸດທີ່ກໍ່ໃຫ້ເກີດການສູນເສຍທັງຊີວິດ ແລະ ຊັບສິນ. ປັດຈຸບັນສະຖານະການອຸປະຕິເຫດຈາກການຈາລະຈອນ ແລະ ຂົນສົ່ງມີແນວໂນ້ມທີ່ສູງຂຶ້ນ ແລະ ບັນຫາການຂັບຂີ່ດ້ວຍຄວາມໄວເກີນຂີດຈຳກັດທີ່ກົດໝາຍກຳນົດ ຍັງເປັນບັນຫາທີ່ເກີດຂຶ້ນທົ່ວໄປໃນສປປ ລາວ ສາຍເຫດເກີດຈາກການຂາດການບັງຄັບໃຊ້ກົດໝາຍທີ່ເຄັ່ງຄັດເຮັດໃຫ້ຜູ້ຂັບຂີ່ລະເມີດການໃຊ້ຄວາມໄວ, ຕະຫຼອດຈົນການກຳນົດຂີດຈຳກັດຄວາມໄວທີ່ບໍ່ເໝາະສົມ ແລະ ຍັງບໍ່ສາມາດນັບຈຳນວນລົດການເຂົ້າອອກຂອງແຕ່ລະເສັ້ນທາງ. ສະນັ້ນ, ການກວດຈັບຄວາມໄວ ແລະ ນັບຈຳນວນລົດຈຶ່ງເປັນໜຶ່ງຫຼັກການມາຊ່ວຍການຈາລະຈອນຂອງລົດຕາມຖະໜົນທົນທາງ ໃຫ້ມີຄວາມປອດໄພ ແລະ ສາມາດຮັບຮູ້ຄວາມໜ້າແໜ້ນຂອງຈຳນວນລົດທີ່ສັນຈອນໄປມາໃນແຕ່ລະເສັ້ນທາງໃນແຕ່ລະມື້.

ດັ່ງນັ້ນ, ພວກຂ້າພະເຈົ້າຈຶ່ງມີແນວຄວາມຄິດຕ້ອງການສຶກສາຄົ້ນຄວ້າ, ເຕັກນິກຂັ້ນຕອນວິທີການກວດຈັບຄວາມໄວ ແລະ ນັບຈຳນວນລົດ ດ້ວຍການນຳໃຊ້ຮູບແບບເຕັກນິກຂອງ CNN (Convolutional neural network) ແລະ YOLO (You Only Look Once) ເພື່ອມາພັດທະນາແມ່ແບບ (Model) ເພື່ອຫຼຸດການໃຊ້ຄືນກວດຈັບຄວາມໄວ ແລະ ນັບຈຳນວນລົດເຂົ້າອອກ.

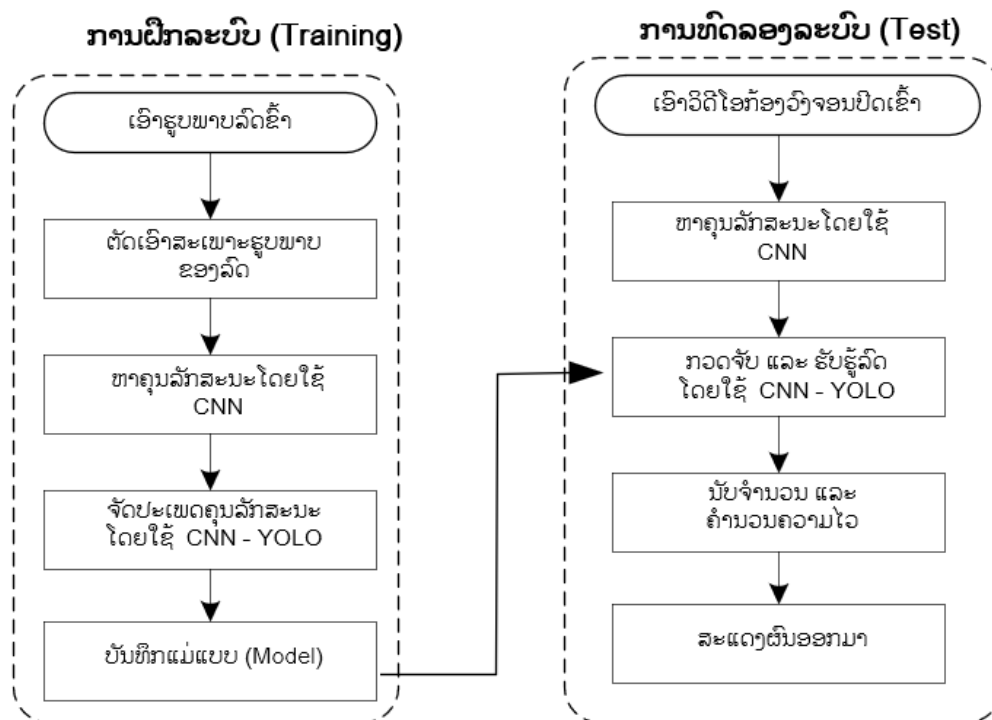
2. ຈຸດປະສົງ

- ເພື່ອສຶກສາວິທີການເຮັດວຽກຂອງ CNN ແລະ YOLO Algorithms ໃນການປະຍຸກໃຊ້ໃນການນັບຈຳນວນລົດຕາມທ້ອງຖະໜົນ ແລະ ປະຍຸກໃຊ້ກັບການກວດຈັບຄວາມໄວຂອງລົດຕາມທ້ອງຖະໜົນແບບເວລາຈິງ (Real-Time)

- ເພື່ອພັດທະນາລະບົບແບບອອນລາຍ ໃນການກວດຈັບຄວາມໄວຂອງລົດ ແລະ ກວດນັບຍານພາຫະນະທີ່ເກີດຂຶ້ນໃນພາບຈາກກ້ອງວົງຈອນປິດໂດຍໃຊ້ CNN ແລະ YOLO algorithms,
- ເພື່ອອຳນວຍຄວາມສະດວກການຈາລະຈອນ ແລະ ຊ່ວຍຫຼຸດຜ່ອນບັນຫາການຈາລະຈອນຕາມທ້ອງຖະໜົນຂອງ ສປປ ລາວ

3. ວິທີດຳເນີນການຄົ້ນຄວ້າ

ວິທີການດຳເນີນການຄົ້ນຄວ້າຂອງການພັດທະນາລະບົບໃນການວິໄຈຄັ້ງນີ້ ປະກອບດ້ວຍ 2 ຂັ້ນຕອນ ຫຼັກຄື ຂັ້ນຕອນການຝຶກອົບຮົມລະບົບ (training step) ເພື່ອສ້າງແມ່ແບບ ແລະ ຂັ້ນຕອນທົດສອບແມ່ແບບຂອງລະບົບ (testing step), ລຳດັບຂັ້ນຕອນຢ່ອຍຂອງແຕ່ລະຂັ້ນຕອນໄດ້ສະແດງໃນ ຮູບທີ 1.



ຮູບທີ 1: ແຜນວາດການເຮັດວຽກຂອງລະບົບ

ສຳລັບຂັ້ນຕອນການສ້າງແມ່ແບບ: ໃນເບື້ອງຕົ້ນແມ່ນລວບລວມແມ່ນຂໍ້ມູນພາບຍານພາຫະນະຕ່າງ, ເອີ້ນວ່າ ກຸ່ມຂໍ້ມູນ (Dataset), ຫຼັງຈາກນັ້ນນຳໃຊ້ວິທີການຂອງ CNN ເພື່ອຄຳນວນຫາຄຸນລັກສະນະຂອງຂໍ້ມູນພາບ, ແລ້ວນຳໃຊ້ວິທີການ CNN-YOLO ເພື່ອຈັດປະເພດຄຸນລັກສະນະຂອງຂໍ້ມູນພາບ, ສຸດທ້າຍບັນທຶກຜົນໄດ້ຮັບ ເພື່ອໃຫ້ເປັນແມ່ແບບ (Model).

ສຳລັບຂັ້ນຕອນການທົດສອບແມ່ແບບ: ການທົດສອບແມ່ແບບແມ່ນນຳໃຊ້ຂໍ້ມູນພາບວິດີໂອຈາກກ້ອງວົງຈອນປິດນຳເຂົ້າສູ່ລະບົບ, ຫຼັງຈາກນັ້ນນຳໃຊ້ວິທີການຂອງ CNN ເພື່ອຄຳນວນຫາຄຸນລັກສະນະຂອງຂໍ້ມູນພາບວິດີໂອ ແຕ່ລະເຟມ, ແລ້ວນຳໃຊ້ວິທີການ CNN-YOLO ເພື່ອກວດຈັບ ແລະ ຮັບຮູ້ຍານພາຫະນະ, ນັບຈຳນວນລົດ ແລະ ຄຳນວນຄວາມໄວ, ສຸດທ້າຍສະແດງຜົນໄດ້ຮັບອອກມາ.

ເຄື່ອງມືທີ່ໃຊ້ສໍາລັບການຄົ້ນຄວ້າ: ການພັດທະນາລະບົບນີ້ແມ່ນໃຊ້ ພາສາ Python ສໍາລັບການປະມວນຜົນຂໍ້ມູນຮູບພາບ, ຂໍ້ມູນພາບວິດີໂອ. ສ່ວນພາສາ HTML ແລະ CSS ໃຊ້ໃນການອອກແບບສ່ວນຕິດຕໍ່ກັບຜູ້ໃຊ້ (User Interface). ສ່ວນອຸປະກອນ Hardware ໃຊ້ຮັບຂໍ້ມູນພາບວິດີໂອແມ່ນກ້ອງ Webcam.

3.1 ການກະກຽມຊຸດຂໍ້ມູນເພື່ອສ້າງແມ່ແບບ

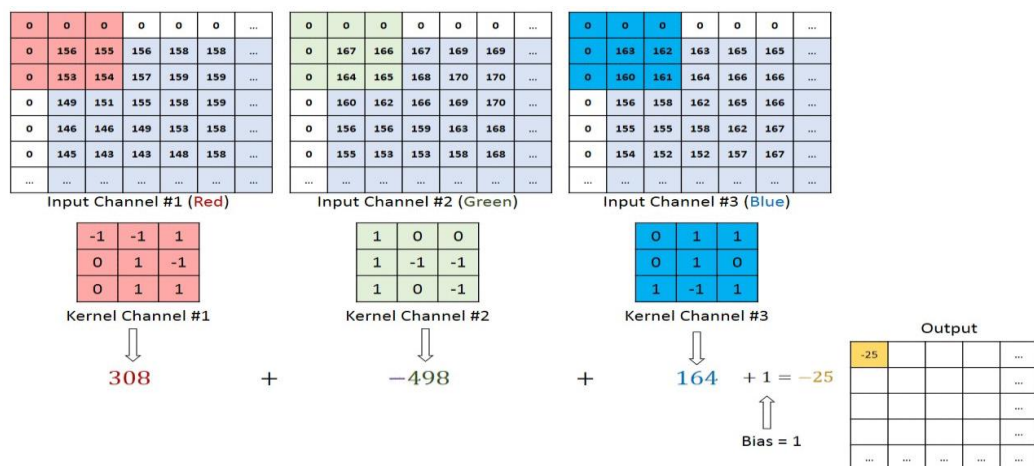
ໃນຊຸດຂໍ້ມູນເປັນຮູບພາບຖ່າຍຈາກກ້ອງຖ່າຍຮູບທົ່ວໄປໂດຍຖ່າຍພາບຈຳນວນຊຸດຂໍ້ມູນຕາມແຕ່ລະປະເພດລົດເຊັ່ນ: ລົດຈັກ 1.000 ຮູບຂຶ້ນໄປ, ລົດໃຫຍ່ 1.000 ຮູບຂຶ້ນໄປ ແລະ ລົດຖີບ 1.000 ຮູບຂຶ້ນໄປ ເປັນຊຸດຂໍ້ມູນສໍາລັບການຝຶກອົບຮົມໃຫ້ລະບົບຮັບຮູ້. ຈາກນັ້ນຈຶ່ງຊອກຫາຄຸນລັກສະນະຂອງລົດແລ້ວບັນທຶກເປັນສາມໄຟຣ໌ .weights ເປັນໄຟຣ໌ເກັບຄຸນລັກສະນະ .cfg ເປັນໄຟຣ໌ເກັບຄ່າ Defaults ຂອງ CNN – YOLO ແລະ .names ເປັນໄຟຣ໌ເກັບຊື່ Labels ຂອງວັດຖຸ



ຮູບທີ 2: ຕົວຢ່າງຊຸດຂໍ້ມູນ

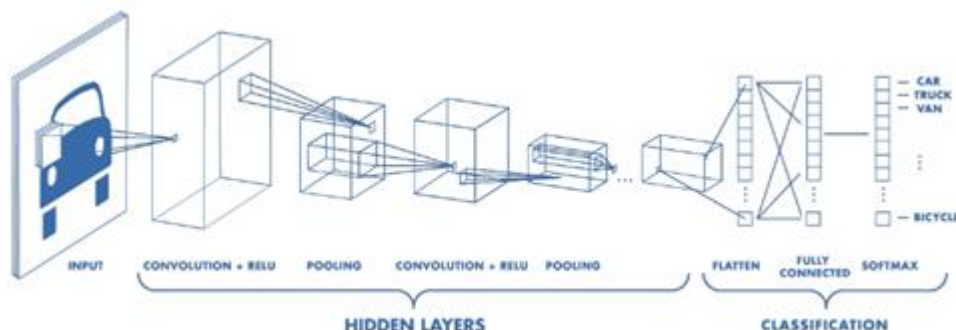
3.2 ໂຄງຂ່າຍປະສາດທຽມແບບຄອນໂວລູຊັນ

ໂຄງຂ່າຍປະສາດທຽມແບບຄອນໂວລູຊັນ (Convolutional Neural Network : CNN) ເຊິ່ງເປັນໂຄງຂ່າຍປະສາດທຽມໜຶ່ງໃນກຸ່ມ Bio-inspired ໃນການເຮັດວຽກຂອງ CNN ເພື່ອການຊອກຫາຄຸນລັກສະນະຂອງຮູບພາບໂດຍການແບ່ງຮູບພາບເປັນຫຼາຍສ່ວນນ້ອຍໆ ຈາກນັ້ນຈຶ່ງຄຳນວນຊອກຫາຄຸນລັກສະນະໂດຍການເລື່ອນໄປເທື່ອລະຈຸດຂອງຮູບພາບ (Prabhu, 2018).



ຮູບທີ 3: ຕົວຢ່າງການຄຳນວນຫາລັກສະນະຂອງວັດຖຸ

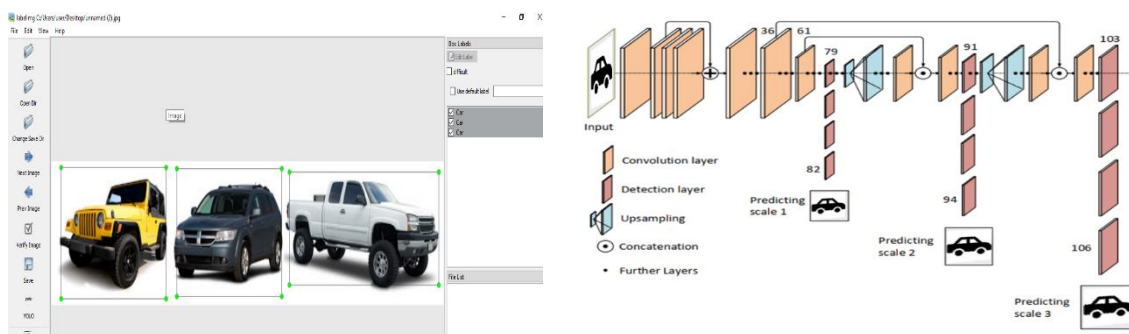
ເຊິ່ງໃນຫຼັກການຂອງ CNN ມີຂັ້ນຕອນຂອງການຄຳນວນຫາຄົ້ນລັກສະນະເຊິ່ງປະກອບໄປດ້ວຍຂັ້ນຕອນ Convolution layers filters (Kernels), Pooling, Fully connected layers (FC) ແລະ SoftMax function ເປັນຕົວຄຳນວນລັກສະນະຂອງວັດຖຸ. (Nattawat.R and Olarik.S, 2018)



ຮູບທີ 4: ໂຄງສ້າງການເຮັດວຽກຂອງ CNN

3.3 You Only Look Once (YOLO)

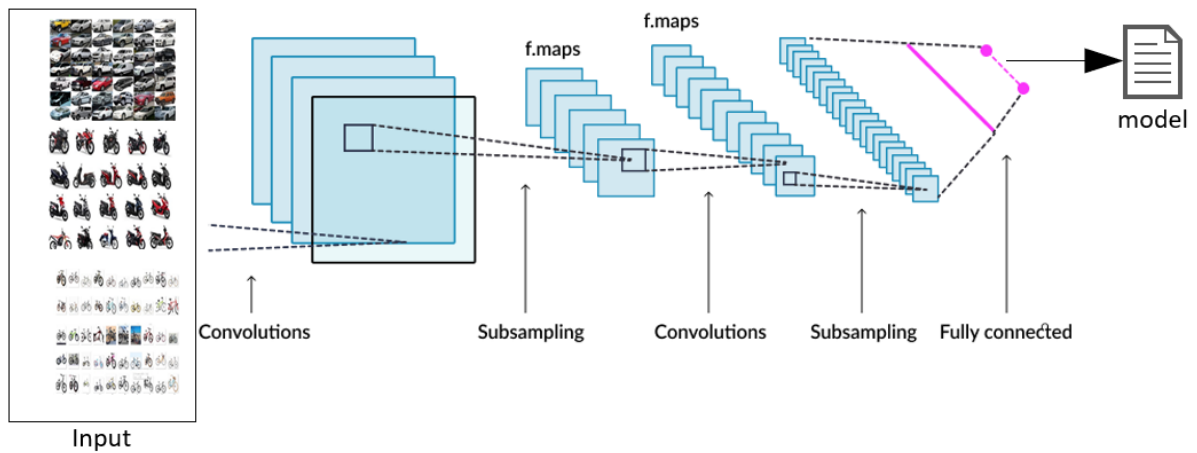
You Only Look Once (YOLO) ເປັນ Algorithms ທີ່ຂະຫຍາຍຫຼັກການຂອງ CNN ໃຫ້ໄວກວ່າ ເກົ່າໂດຍການນຳໃຊ້ວິທີການ Classification ແລະ ການຊອກຫາຕຳແໜ່ງຂອງວັດຖຸໂດຍໃຊ້ Bounding Box ເຂົ້າໄປຊ່ວຍແລ້ວແບ່ງພາບອອກເປັນຫຼາຍສ່ວນນ້ອຍໆ ເພື່ອຄຳນວນຊອກຫາຄຸນລັກສະນະໄປພ້ອມໆກັນ ເຊິ່ງອົງປະກອບ ແລະ ວິທີການເຮັດວຽກຂອງ YOLO ໂດຍຂະບວນການສອນປະກອບດ້ວຍ 3 ຂັ້ນຕອນຄື: 1. ຕ້ອງມີຮູບວັດຖຸເພື່ອນຳໄປໃຊ້ໃນການສອນລະບົບໃຫ້ລະບົບຮັບຮູ້ຂໍ້ມູນເກັບເປັນແມ່ແບບ (Model), 2. ກຳນົດ ບ້າຍກຳກັບໃຫ້ວັດຖຸຮູບພາບ (Image Annotations) ແລະ 3. ນຳເອົາຊຸດຂໍ້ມູນເຂົ້າໃນລະບົບໂດຍໃຊ້ວິທີການ ໂຄງຂ່າຍປະສາດທຽມ (Convolutional Neural Network: CNN) ແລ້ວຫາຄຸນລັກສະນະຂອງຊຸດຂໍ້ມູນ ແລ້ວເກັບເປັນແມ່ແບບ (Model) ໄວ້ໃຫ້ລະບົບຮັບຮູ້ຊຸດຂໍ້ມູນ. (Sitthitouch.S, 2017)



ຮູບທີ 5: ຂັ້ນຕອນໂຄງສ້າງການເຮັດວຽກຂອງ YOLO

3.4 ວິທີການສ້າງແມ່ແບບ (Creating Model)

ການສ້າງແມ່ແບບເພື່ອເອົາຂໍ້ມູນຂອງລົດໄປສອນໃຫ້ລະບົບຮັບຮູ້, ແມ່ນການເອົາຊຸດຂໍ້ມູນທັງໝົດຂອງ ລົດໄປເຂົ້າລະບົບດ້ວຍການນຳໃຊ້ຫຼັກການຂອງ CNN – YOLO ເພື່ອສະກັດເອົາຄຸນລັກສະນະຂອງຮູບພາບ ລົດແຕ່ລະປະເພດແລ້ວເອົາໄປເກັບໄວ້ເປັນ ໄຟລ ແມ່ແບບ (Model) ເພື່ອຈະນຳໄປປຽບທຽບກັບຂໍ້ມູນຈາກ ກ້ອງວິດີໂອທີ່ໄດ້ຮັບ ເຊິ່ງຂັ້ນຕອນການເຮັດວຽກຂອງລະບົບມີດັ່ງລຸ່ມນີ້.



ຮູບທີ 6: ຂັ້ນຕອນວິທີການຝຶກໃຫ້ລະບົບຮັບຮູ້

ອະທິບາຍຄວາມໝາຍແຕ່ລະຂັ້ນຕອນວິທີການຝຶກໃຫ້ລະບົບຮັບຮູ້:

- Input ໝາຍເຖິງການສົ່ງຮູບພາບເຂົ້າໄປໃນລະບົບເພື່ອພັດທະນາຮູບແບບ
- Convolutions ເປັນຂັ້ນຕອນທີ່ອີດສຳລັບການແຍກຄຸນລັກສະນະຂອງຮູບພາບທີ່ນຳເຂົ້າ ແລະ ຮັກສາຄວາມສຳພັນລະຫວ່າງ Pixels ດ້ວຍການຮຽນຮູ້ຄຸນລັກສະນະຂອງຮູບພາບ ໂດຍໃຊ້ຂໍ້ມູນພາບແຍກເປັນໜ້າຕ່າງນ້ອຍໆ, ແລ້ວສະກັດເອົາຄຸນລັກສະນະໂດຍການຄຳນວນທາງຄະນິດສາດທີ່ໃຊ້ການຄຳນວນຄ່າ $H*W*D$ (H = Height, W = Width, D = Dimension)
- Subsampling ຫຼື Pooling ເປັນການເອົາອົງປະກອບໃຫຍ່ສຸດທີ່ຫາໄດ້ຄຸນລັກສະນະມາຈາກຂັ້ນຕອນ Convolutions ໂດຍການກຳນົດຕົວກັນຕອງ (Filters) ຂະໜາດ 2×2
- Fully connection ເຮັດໜ້າທີ່ສຳລັບການຮຽນຮູ້ ແລະ ຈຳແນກປະເພດຂອງລົດຕາມຈຳນວນລົດ
- Model ເປັນການເອົາຄຸນລັກສະນະທີ່ສະກັດໄດ້ແລ້ວໄປເກັບໄວ້ເປັນໄຟຣ໌ແມ່ແບບ (Model)

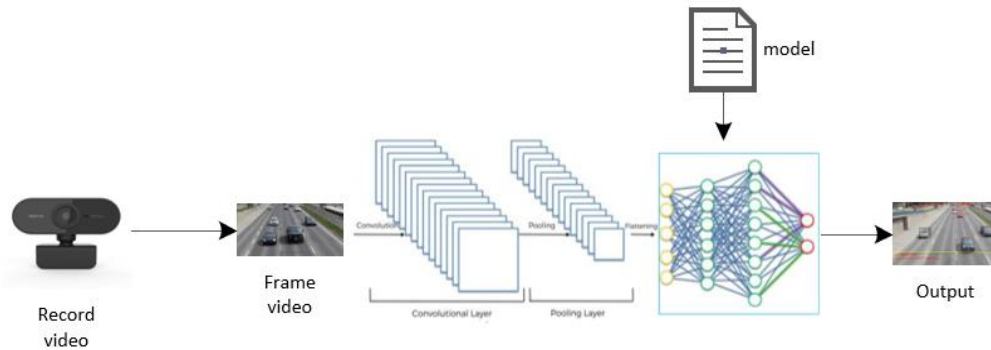
3.5 ວິທີການທົດລອງເອົາຂໍ້ມູນຮູບພາບວິດີໂອໄປຜ່ານລະບົບແລ້ວກວດຈັບລົດ

ເປັນການນຳເອົາຂໍ້ມູນຮູບພາບວິດີໂອໄປຜ່ານລະບົບແລ້ວເຮັດການກວດສອບດ້ວຍການສະກັດເອົາຄຸນລັກສະນະແຕ່ລະ Frame ຂອງວິດີໂອ ແລ້ວເອົາໄປປຸງປັບທຽບກັບຄຸນລັກສະນະທີ່ມີເກັບໄວ້ໃນ Model ແລ້ວສະແດງຜົນອອກມາ.

ອະທິບາຍຄວາມໝາຍແຕ່ລະຂັ້ນຕອນວິທີການພັດທະນາແບບທົດລອງ:

- Record video ໝາຍເຖິງກ້ອງຖ່າຍຮູບເກັບເປັນວິດີໂອແລ້ວສົ່ງເປັນ Frame ເຂົ້າໃນລະບົບ
- Frame Video ໝາຍເຖິງວິດີໂອທີ່ສົ່ງເຂົ້າໄປປະມວນຜົນໂດຍໃຊ້ Convolutions
- Convolutions ເປັນຂັ້ນຕອນທີ່ອີດສຳລັບການແຍກຄຸນລັກສະນະຂອງແຕ່ Frame Video ທີ່ນຳເຂົ້າ ແລະ ຮັກສາຄວາມສຳພັນລະຫວ່າງ Pixels ດ້ວຍການຮຽນຮູ້ຄຸນລັກສະນະຂອງຮູບພາບໂດຍໃຊ້ຂໍ້ມູນພາບແຍກອອກເປັນໜ້າຕ່າງນ້ອຍໆ ແລ້ວສະກັດເອົາຄຸນລັກສະນະໂດຍການຄຳນວນທາງຄະນິດສາດທີ່ໃຊ້ໃນການຄຳນວນຊອກຫາຄ່າ $H*W*D$ (H = Height, W = Width, D = Dimension)
- Pooling ເປັນການເອົາອົງປະກອບຂອງຄຸນລັກສະນະຄ່າໃຫຍ່ສຸດທີ່ຫາໄດ້ຈາກຄຸນລັກສະນະມາຈາກຂັ້ນຕອນ Convolutions ໂດຍການກຳນົດຕົວກັນຕອງ (Filters) ຂະໜາດ 2×2

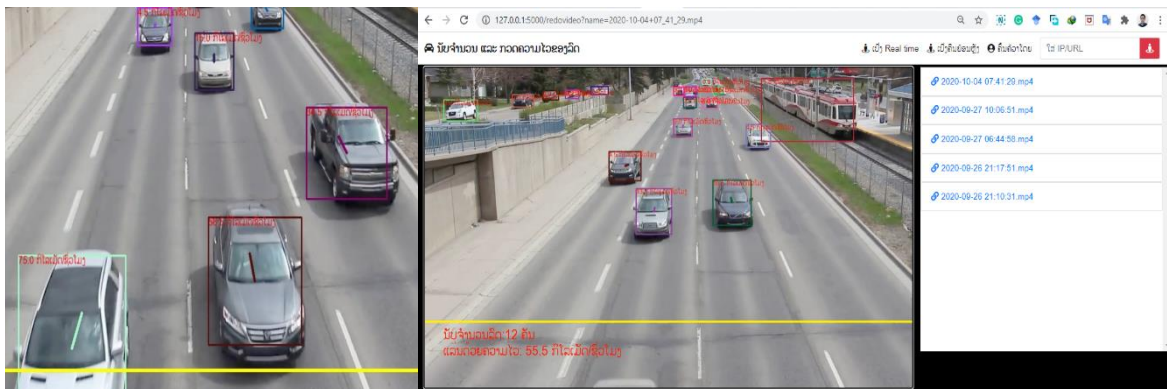
- Fully connection ແລະ Model ເຮັດໜ້າທີ່ສໍາລັບການຮຽນຮູ້ ແລະ ຈຳແນກປະເພດຂອງລົດໂດຍການເອົາຂໍ້ມູນແຕ່ Frame ວິດີໂອໄປປຽບທຽບກັບຂໍ້ມູນທີ່ມີໃນ Model



ຮູບທີ 7: ຂັ້ນຕອນວິທີການທົດລອງ

4 ຜົນການຄົ້ນຄວ້າ

ຜົນການຄົ້ນຄວ້າວິໄຈຄັ້ງນີ້ແມ່ນສຶກສາວິທີການພັດທະນາແມ່ແບບ (Model) ເພື່ອນຳເອົາຂໍ້ມູນຮູບພາບລົດເຂົ້າໄປຝຶກອົບຮົມ, ກວດຈັບຄວາມໄວ ແລະ ນັບຈຳນວນໄດ້ດ້ວຍການກວດຈັບຜ່ານກ້ອງຖ່າຍວິດີໂອ ຫຼື ໄຟຣ໌ວິດີໂອຜ່ານທາງ Web Brower ອິນເຕີເນັດ (Internet) ໄດ້.



ຮູບທີ 8: ຜົນການກວດຈັບ

5 ສະຫຼຸບຜົນການຄົ້ນຄວ້າ ແລະ ຂໍ້ສະເໜີ

5.1 ສະຫຼຸບຜົນການຄົ້ນຄວ້າ

ຈາກການຄົ້ນຄວ້າຄັ້ງນີ້ເຫັນວ່າການເອົາຂໍ້ມູນລົດເຂົ້າໄປພັດທະນາເປັນແມ່ແບບແມ່ນສາມາດເຮັດໄດ້ໂດຍການໃຊ້ເວລາໃນການຝຶກໃຫ້ລະບົບຮັບຮູ້ ແມ່ນໃຊ້ເວລາປະມານ 2 ຫາ 3 ຊົ່ວໂມງ (ອີງຕາມປະສິດທິພາບຂອງເຄື່ອງຄອມພິວເຕີ).

ສໍາລັບການທົດລອງການກວດຈັບຄວາມໄວ, ນັບຈຳນວນລົດໂດຍເອົາຂໍ້ມູນວິດີໂອມາປະມວນຜົນໃຊ້ເວລາປະມານ 0,2 ວິນາທີຕໍ່ Frame ແລະ ສາມາດກວດຈັບຄວາມໄວ, ນັບຈຳນວນລົດໄດ້ສະເພາະລົດທີ່ຫັນໜ້າເຂົ້າຫາໜ້າກ້ອງເທົ່ານັ້ນ ໂດຍຜ່ານການທົດລອງແມ່ນສາມາດກວດຈັບໄດ້ໂດຍຄ່າສະເລ່ຍປະມານ 90% ຈາກກ້ອງຖ່າຍວິດີໂອ Webcam ຫຼື ໂທລະສັບ.

5.2 ຂໍ້ສະເໜີ

ຜ່ານການສຶກສາວິໄຈຄັ້ງນີ້ຍັງມີຂໍ້ຄົງຄ້າງເລື່ອງການນັບຈຳນວນລົດ ແລະ ຈັບຄວາມໄວຂອງລົດທີ່ມາຈາກທິດທາງກົງກັນຂ້າມຂອງໜ້າກ້ອງຖ່າຍວິດີໂອ ແລະ ຍັງບໍ່ສາມາດນັບຈຳນວນລົດຕາມປະເພດລົດໄດ້. ສະນັ້ນ, ສໍາລັບການສຶກສາຄົ້ນຄວ້າຕໍ່ໜ້າ ກໍ່ສາມາດເຮັດໄດ້ໂດຍການຜັນຂະຫຍາຍຕໍ່ນໍາໃຊ້ເຕັກນິກ ແລະ ວິທີການອື່ນເພື່ອໃຫ້ສາມາດກວດຈັບລົດ ແລະ ຈຳແນກປະເພດລົດ ແລະ ນັບຈຳນວນລົດຕາມປະເພດໄດ້.

ເອກະສານອ້າງອີງ

- Cesar G. Pachón-Suescún, Javier O. Pinzón-Arenas, Robinson Jiménez-Moreno. (2019) **Detection of Scratches on Cars by Means of CNN and R-CNN**
- Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. (2016). **Real-Time Object Detection**. IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- Prabhu (Mar 4, 2018). **Understanding of Convolutional Neural Network (CNN) Deep Learning**. Retrieved from: <https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148>
- Nattawat.R, Olarik.S. (2018). **Comparative Study Between Local Descriptors and Deep Learning for Silk Pattern Image Retrieval**. Retrieved from: http://research.msu.ac.th/msu_journal/upload/articles/article2392_99979.pdf
- Natthapat.S (2019). **Traffic Signs Detection System by Using Deep Learning**. Retrieved from: <http://58.181.147.25/ojsjournal/index.php/miniconference/article/view/2147/2001>
- Sitthitouch.S. (2017). **Generating an Architectural Model Using Object Recognition to Observe Human Movement in Public Space Case Study: 3-dimensional Parametric Model of Bus Station Platforms**. Retrieved from: <http://ias.it.msu.ac.th/course/1201374-Image-processing/1-2562/Project-Document/Group-4-Object-Detection-using-YOLO-V3.pdf>
- Zhong-Qiu Zhao, Member, Peng Zheng, Shou-tao Xu, and Xindong Wu (2019). **Object Detection with Deep Learning**. Retrieved from: <https://arxiv.org/pdf/1807.05511>