WHITE PAPER

Diabetic Retinopathy Detection Powered By Intel AI

Authors:

1. Gina Mathew (Technical Lead)
2. Sindhu Ramachandran S (Principal Architect)
3. Suchithra VS (Software Engineer)

QuEST Global

Contents

Summary

This paper discusses a compute intensive AI solution, which when connected to a portable fundus camera can be used to classify fundus images into its severity grading for detecting Diabetic Retinopathy. A digital fundus camera is used to take an image of the fundus — the back portion of the eye that includes the retina, macula, fovea, optic disc and posterior pole. When deploying this diabetic retinopathy AI solution, Intel processors are used to power the workloads all the way from Edge to the Server. For deployment of the application in Hospitals, high end processors like the Intel® Xeon® Scalable Processors is preferred for dynamic training and for keeping a huge patient database, while on the contrary in rural area retinal examination, a portable low powered, cost effective UP2 board (Intel Intel Atom® or Intel® Core™ Processors connected with fundus camera) can be deployed. Our deep learning model has a sensitivity of 91.5% and a specificity of 97.86%.

Introduction

Diabetic Retinopathy (DR) is the most common microvascular complication in diabetes. Retinal imaging is the most widely used method of retinopathy screening due to its high sensitivity in detecting retinopathy. The evaluation of the severity and degree of retinopathy associated with a person having diabetes, is currently performed by medical experts based on the fundus or retinal images of the patient's eyes.

Currently, about one in six people with diabetes in the world is from India, setting India's rank as second, in the countries among people suffering from diabetes, with an estimated 77 million diabetic patients. In India, there are an estimated 9,000 to 10,000 ophthalmologists, which forms ratio of one ophthalmologist to 100,000 population. This is well below the World Health Organization's recommended ratio of 1 ophthalmologist per 20,000 population. This shows the serious shortage of trained medical practitioners. Lack of ophthalmologists makes diagnosis difficult in rural areas of India. Also, detecting retinopathy is a time-consuming process and is difficult even for an ophthalmologist to evaluate and examine digital color fundus photographs of retina. Most of the AI solutions for detecting Diabetic Retinopathy, available in market, are cloud deployed making rural enablement difficult.

To resolve these challenges, QuEST aimed to create an AI solution targeting rural areas where connectivity is poor (With Edge device like AAEON's UP2 board powered with Intel Atom/Pentium processor) whereas the big hospitals and health care centers in urban areas can opt for this solution with a higher configuration utilizing Intel Xeon Processors on their premises.

We used data from Kaggle 2019 Challenge hosted by Aravind Hospital in India and retinal images provided by EyePACS from Kaggle 2015 challenge for training our Deep Learning model. In this work, we propose a solution to diagnose Diabetic Retinopathy, leveraging Intel Distribution of OpenVINO Toolkit.

Pitfalls With The Current Solution

Numerous solutions employing AI have come forth for DR diagnosis. But these solutions are difficult to deploy in rural areas since most of the solutions require high end machines/Cloud which is inaccessible in rural areas. Lack of connectivity along with latency and data security issues are some of the other challenges that arise when using those solutions.

Challenge

Fundus image dataset is noisy with variations of retina among people in different geographies. Some of the images were overexposed, underexposed, or were out of focus leading to non-uniform illumination and blurring. Often inter class variations were very small especially among adjacent class making distinguishing of class difficult even for a trained ophthalmologist. The images (from the Kaggle Challenge) were gathered from multiple clinics using a variety of cameras over an extended period of time, which will introduce further variation. Another problem faced was the high imbalance in data. Normal class data outnumbered data from other classes.

Model Training Workflow

Input Dataset

Dataset consisting of fundus images rated by a clinician on a scale of 0 to 4 was used according to the following scale 0->Normal, 1->Mild DR, 2->Moderate DR, 3->Severe DR and 4->Proliferative DR. Although the objective was to detect diabetic retinopathy among natives of India, since the input dataset hosted in the Kaggle 2019 challenge was small we used retinopathy images from the Kaggle 2015 challenge also for training the model to learn the retinopathy features better.

Table 1 and Table 2 show the image count taken for training/validation/testing across each of the different classes for both the datasets. Testing the model was done on the test data set apart from the Kaggle 2019 dataset.

Train/Val/Test	**Class 0** **Normal**	**Class 1** **Mild**	**Class 2** **Moderate**	**Class 3** **Severe**	**Class 4** **Proliferative**
Train	25810	2443	5292	873	708
Test	39533	3762	7861	1214	1206

Table 1: Kaggle 2015 Image count

Train/Val/Test	**Class 0** **Normal**	**Class 1** **Mild**	**Class 2** **Moderate**	**Class 3** **Severe**	**Class 4** **Proliferative**
Train	1166	235	630	122	190
Val	292	59	158	31	47
Test	347	76	211	40	58

Preprocessing

Since all the images were of different size, we had to rescale that to same value. Pre-processing consisted of finding an approximate radius of the retina and scaling down the image. Each image was thus cropped to a square shape with tightly contained circular area of fundus. Since fundus images were captured in different lighting conditions, to compensate for lighting variations, and to highlight the important features in the image, weighted addition of original and Gaussian blurred version of the image was carried out. Different augmentation techniques were also applied to augment the data for creating different input variations. A patch of 256x256 was drawn from the center and used for training the DL model.

Model Training

We used PyTorch framework to train the model. Model was trained using Kaggle 2015 and 2019 datasets. EfficientNet-B4 architecture was used for model training and the loss function used was MSE. Diabetic Retinopathy problem has significance in the ordering of labels i.e. Severe DR can be more detrimental than moderate DR, but less detrimental than Proliferative DR. Since in classification problem, ordering of the labels is discarded, we treated the problem as regression and used a pre-defined threshold to capture the class id. Optimizer used was Rectified ADAM. Rectified Adam (RAdam), is a variant of the Adam optimizer, as it "rectifies" the variance/generalization issues apparent in other adaptive learning rate optimizers. Please refer Table 3 for training summary. We completed training in eight levels of 30 epoch each, beyond which training accuracy didn't improve. Activation layer of EfficientNet-B4 is swish activation. Output of this activation layer was set as a threshold to obtain the result.

Training Summary

Model	EfficientNet-B4
Loss function	MSE
Optimizer	RAdam
LR Scheduler	Cosine Annealing
Batch size	32
Number of Epoch	240

Table 2 Model Details

Solution Details	
Framework	PyTorch 1.2.0
Tools/Library	OpenVINO R2020.1
OS	Ubuntu 16.04 LTS
Topology	EfficientNet-B4

Table 3 Configuration Environment Details

Cross Validation

In order to assess the model, 5-fold cross validation was done on 2019 data. Validation accuracy for 2019 data after 5-fold cross validation was 92.5%.

QuEST Solution Accelerated by Intel Distribution of OpenVINO Toolkit

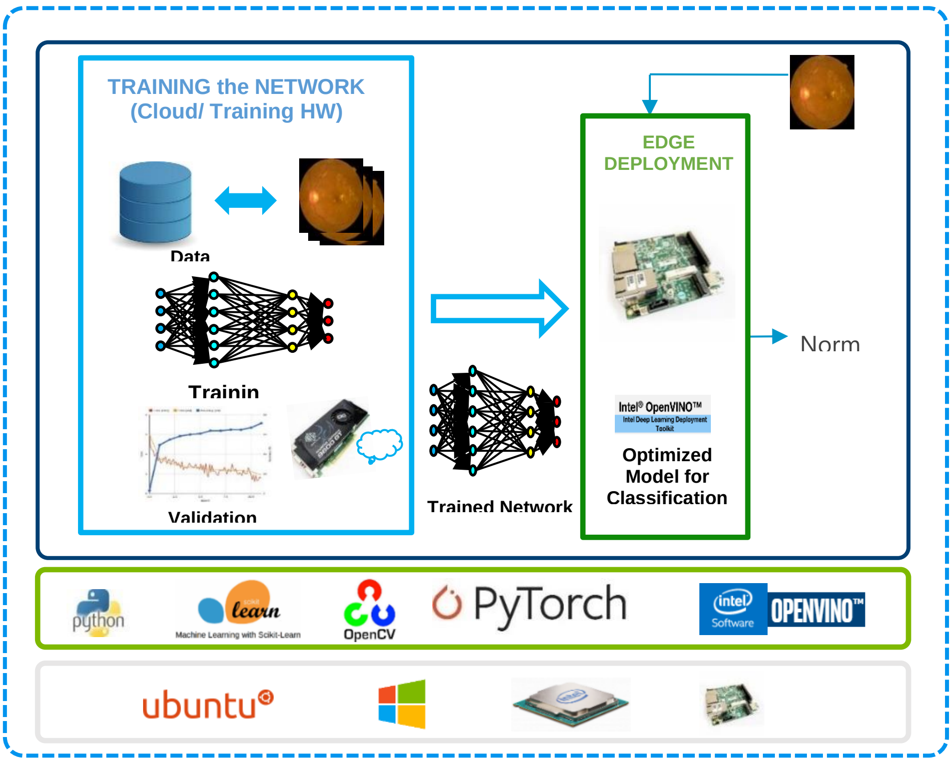

Figure 1 Workflow diagram of our solution

Taking cue from OpenVINO toolkit we accelerated inference of our DL model for Diabetic Retinopathy Classification, to employ model inferencing even in low end Intel embedded processors without compromising in performance.

Figure 1 Workflow diagram of our solution shows the workflow diagram of our solution. As shown in the Figure, training was done offline in a HW platform.

EfficientNet-B4 PyTorch model needs to be converted first to ONNX model. The ONNX model is then converted to OpenVINO IR model using model optimizer. Once the model is converted to IR format this can be read using Inference Engine of OpenVINO. Our application uses Inference Engine APIs to read the model and classify the images.

Rural Area Deployment

The entire rural area deployment use case has been depicted in the Figure 2 given below. Currently, portable Fundus camera/smartphones enabled for capturing Fundus images are available. Fundus images captured using portable fundus camera / smartphone camera can be connected to the AI powered AAEON's UP2 board. This edge device should contain the model pre-trained and optimized for deployment. Hence during inference, on giving an image the model would output the class to which the fundus image belongs as the output. Our application provides an easy interface for anyone to analyse retinal scans, and to make the DR screening process faster and efficient. And based on the inference, those patients can be referred to the hospital wherein further diagnosis and treatment can be prescribed by the ophthalmologist.

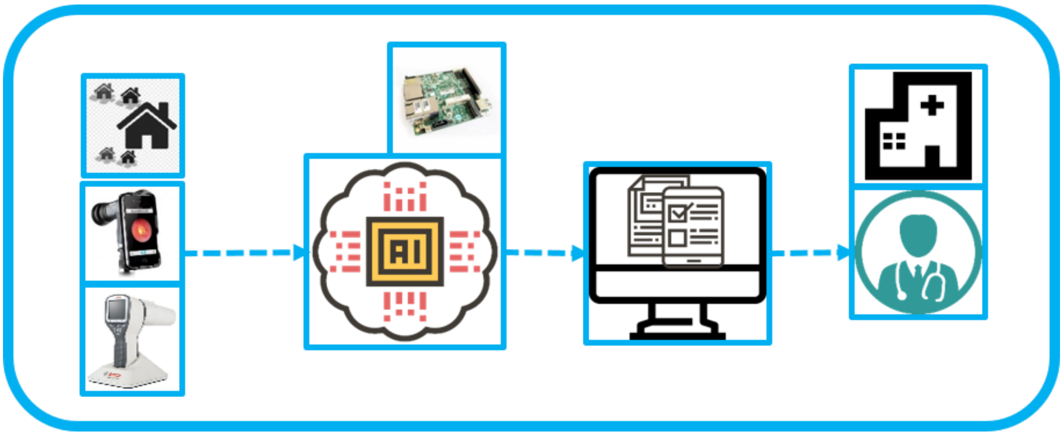

Figure 2 Rural Area Deployment Use Case

Intel Distribution of OpenVINO™

Intel Distribution of OpenVINO toolkit quickly deploys applications and solutions that emulate human vision. Based on Convolutional Neural Networks (CNN), the toolkit extends computer vision (CV) workloads across Intel hardware, maximizing performance. Intel Distribution of OpenVINO toolkit enables deep learning inference from edge to cloud. It accelerates AI workloads, including computer vision, audio, speech, language, and recommendation systems, with support for heterogeneous execution across various Intel hardware such as Intel CPU, Intel Integrated Graphics, Intel Movidius Neural Compute Stick, Intel Neural Compute Stick 2, and Intel Vision Accelerator Design with Intel Movidius VPUs. During CPU execution, OpenVINO uses Intel MKL-DNN for the major primitive acceleration. Internally Inference Engine in OpenVINO, uses Intel TBB as a parallel engine.

Results

Test images set apart from Kaggle 2019 data was fed as input to Intel Distribution of OpenVINO optimized model to run on various Intel Architecture devices ranging from embedded (Intel Atom) to server segment (Intel Xeon Gold 6148) of processors. Each of the input test images are high resolution images. These images are pre-processed and fed to the pre-trained OpenVINO-optimized EfficientNet-B4 model for inference, and experimentation was done on batch size varying from 1 to 32. Figure 3 shows the image/sec throughout on executing PyTorch and OpenVINO optimized model in various IA processors. Execution in Intel Atom CPU processor took about 0.22 s, while the same model on Xeon Gold 6148 processor took approximately 0.009s for batch size 32. Also, it is found that there is a performance gain of 4.49X when using OpenVINO compared to Pytorch on Intel Atom x7-E3950 CPU, and a performance gain of 18.9X when using openvino compared to Pytorch on Intel Xeon Gold 6148 CPU. In terms of model accuracy, our model gave sensitivity of 91.5% and specificity of 97.8%.

Table 4 Model Performance contains performance metric of our model.

Metric	Score
Validation accuracy	92.5%
Test Accuracy	91.5%
Kappa score	96%
Sensitivity	91.5%
Specificity	97.86%
Precision score	92%

Table 4 Model Performance contains performance metric of our model

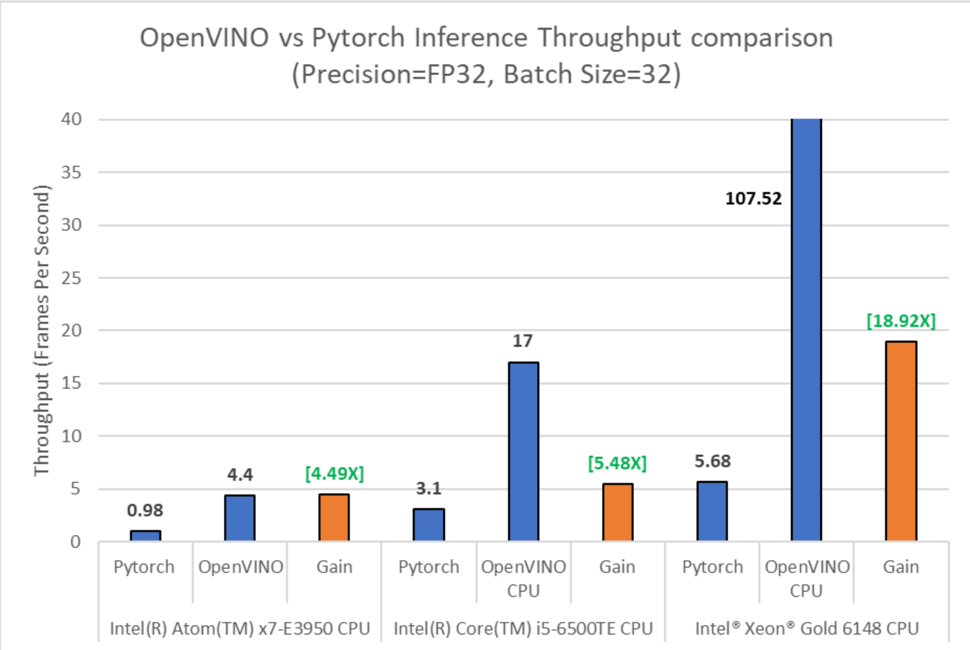

Figure 3 OpenVINO Vs PyTorch Inference

Conclusion

With the available state-of-the-art deep learning technology, QuEST has trained DL model using EfficientNet-B4 architecture, to come up with an AI based solution that can be used on edge and server side. Since optimization of Deep learning model has been done using Intel Distribution of OpenVINO, performance in edge has not been heavily compromised. The hardware board (UP2) can be connected with a handheld fundus camera to provide a portable solution when rural area eye camps are organized.

Thanks to the increased performance and workload flexibility provided by Intel architecture and Intel software, clinicians are better able to examine and diagnose more patients in a broader set of geographies. This allows for more individuals throughout India to hopefully gain a timelier diagnoses and improved treatment for their diabetic retinopathy.

It can also be made possible to transfer medical records (with fundus images) from multiple edge devices to server which enables archiving rare condition images for further medical inspection and studies. For deployment of the application in Hospitals, high send processors like Intel Xeon is preferred. Thus, this benchmarking enables to position Intel processor in different segment for compute intensive Diabetic retinopathy application in Edge and Server side. During deployment, when new image patterns arise, those images can be sent to the Xeon processor where we can store and re-train our model with further images, for boosting the performance. There is scope for further performance enhancement with INT8 quantization and will be executed in the future.

Appendix A

Hardware Configuration Details

	Intel(R) Core(TM) i5-6500TE CPU	**Intel(R) Atom(TM) x7-E3950 CPU**	**Intel(R) Xeon(R) Gold 6148 CPU**
Test by	QuEST-Global	QuEST-Global	QuEST-Global
Test date	16-04-2020	16-04-2020	16-04-2020
Platform	x86_64	x86_64	x86_64
# Nodes	1	1	2
# Sockets	1	1	2
CPU	I5-6500TE	E3950	6148
Cores/socket, Threads/socket	4/4	4/4	20/40
Serial No cpu0			
Serial No cpu1			
ucode			
HT	No	No	Yes
Turbo	Yes	No	Yes
BIOS version (including microcode version: cat /proc/cpuinfo | grep microcode –m1)			0x200005e
System DDR Mem Config: slots / cap / run-speed	DDR4-1866/2133, DDR3L-1333/1600 @ 1.35V	DDR3L (ECC and Non ECC) up to 1866MT/s; LPDDR4 up to 2400 MT/s	DDR4-2666 MHz
System DCPMM Config: slots / cap / run-speed	-	-	
Total Memory/Node (DDR+DCPMM)	8 GB	4 GB	192 GB
Storage - boot			
Storage - application drives			
NIC			
PCH			
Other HW (Accelerator)			
OS	Ubuntu 18.04.2 LTS	Ubuntu 18.04.2 LTS	Ubuntu 18.04 Bionic
Kernel	4.15.0-88-generic	4.15.0-88-generic	4.15.0-50-generic

Authors Bio

Gina Mathew is a Technical Lead at QuEST Global. She is experienced in developing and implementing deep learning solutions for various domains using OpenVINO on Intel Architecture based platforms.

Ms. Sindhu Ramachandran S is a Principal Architect at QuEST Global. She is currently involved in algorithm implementation and optimization in the Medical Devices and Hi-Tech Domains. Her key area of interest is image processing and machine learning. She leads the artificial intelligence (AI) / deep learning (DL) initiative at QuEST Global and provides her expertise in project activities related to AI & DL.

https://www.linkedin.com/company/quest-global

https://www.facebook.com/QuESTGlobal

https://twitter.com/QuEST_Global

https://www.youtube.com/user/questglobal10

About QuEST Global

For more than 20 years, QuEST Global has aimed to be a trusted global product engineering and lifecycle services partner to many of the worlds' most recognized companies in the Aero Engines, Hi-Tech, Aerospace & Defense, Transportation (Auto and Rail), Power and Industrial, Oil & Gas and Medical Devices industries. With a global presence in 14 countries, 67 global delivery centers and 12,000+ personnel, QuEST Global believes that it is at the forefront of the convergence of the mechanical, electronics, software and digital engineering innovations to engineer solutions for a safer, cleaner world. QuEST Global's deep domain knowledge and digital expertise aim to help its clients accelerate product development and innovation cycles, create alternate revenue streams, enhance consumer experience and make manufacturing processes and operations more efficient.

BORN TO ENGINEER

quest-global.com