Virtual Hospital with Real-Time Image Diagnosis

Habtamu Demeke Mihertie and Ying Gao
School of Marine Science and Technology, Northwestern Polytechnical University, China.
Email: derui29@yahoo.com

Abstract. Recently, deep learning models such as deep convolutional neural networks have shown great success in different imaging tasks in general and in detection of diabetic retinopathy in particular. An automated detection system is undoubtedly a great help for the screening stage of diabetic retinopathy for further treatment to reduce the burden on the public health system. In this paper, a general framework for automated detection virtual environment is developed as a virtual hospital using a convolutional neural network (CNN) approach as imaging end. Image classification is done in two stages. The best model being 87.12% accurate for the first stage on more than 53000 images which is better than previous works tested only on very less amount of test data. Using this model, a standalone application is generated with extra features for user interaction to be used by the patient in the virtual hospital when he presses a key on the keyboard.

1. Introduction
Diabetes, in general, is a disorder of sugar metabolism in our body. This silent disease associated with an increased likelihood of suffering systemic vascular complications, including stroke, coronary heart disease, and heart failure. Though diabetes affects the eyes in many ways, diabetic retinopathy (DR) is the most common and serious complication. More than thirty percent of diabetic patients develop DR, which is the foremost cause of preventable blindness in working age people. The worst thing of DR is that it can be unnoticed by the patient until the retina has been damaged to a level where treatment is nearly impossible. Therefore, early detection, performed by a periodic screening of the eye fundus, is fundamental to treat DR and decrease the probability of suffering blindness at an early age. However, the high number of potential patients and the need for periodic inspection is a huge burden for the public health units and therefore automatic image processing applications are needed to reduce the cost associated with the diagnosis of this illness and most importantly increase the accuracy of diagnosis.

From the application’s point of view, a lot has been done focusing on the development of application software and simulators most of which start designing 3D models of different organs. Robb’s survey [1] on several types of multidimensional image data-sets, and several examples of real clinical applications described using 3D, 4D and 5D fused image data-sets and VR technology for image-guided Interventions, image-guided surgery, and image-guided therapy. Eder Govea et al. [2] used the blender, open source software, to design low-cost VE using the Python language in Windows operating system depicting four different cases. They also developed a high-quality virtual scenario in 2018 [3] using almost the same methodology to design and develop with a high degree of realism for medical applications such as surgery planning, simulation, and training.

Filippi S. et al. presents the 3D modeling of a femur by using the software called 3ds Max.[4] work on X-ray images, representing the specific patient’s anatomy. The script does not implement image
segmentation; contours used in the case studies have been obtained manually using the software package Matlab.

We can see from the previous researches that most of the applications of VE deal with either developing interactive software for the information system, software development to make the medication procedures faster and more efficient, or simulators for training and educational purpose. Although some involve a certain level of intelligence by incorporating decision support for well-known and common symptoms using vital signals and demographic data, as The Global Virtual Hospital developed by Petrasek D. et al. [5], the systems can be made fully automated and independent by including automated medical diagnosis applications. However, to increase the level of realism and benefit both sides (health care units and patients), we need to rely on new technologies like virtual reality and use of devices as input modalities from the patients’ side. We can either use telemedicine to get patient data to the central unit or deploy the diagnosis application at the far end for the last decisions. Singh, R. and A. K. [6] investigate the basic principles of such systems, tele-immersive systems, with its main components and development history.

DR has many levels but can be classified into two categories as nonproliferative and proliferative. Nonproliferative DR is less critical as its progression can still be kept under control whereas the latter is the most vital and dangerous stage since it is the last stage before blindness. Therefore, it is vital if we can detect DR when it is in its first stages, that is what we can do, prevention. However, it’s more challenging because some of the signs are difficult to detect than during the proliferative stage.

There are many methods related to the detection of DR ranging from the easiest ones like detection of early signs for early stages to the full automated detection of all the levels using deep learning. Deep learning employs different algorithms and techniques to detect candidates of being a lesion, and then rely on the use of a classifier of different kinds or a multilayer perceptron neural network to make the final assessment.

Bengio et al. [7] provided a thorough review of popular techniques to learn features used before the breakthrough of Alex Net which are considered as classical approaches. Another dedicated review on the application of deep learning to medical image analysis is published by Shen et al. [8] focusing on the success of deep learning methods in medicine.

From image classification’s point of view, Krizhevsky et al., [9] introduced Alex Net in 2012 which became famous when it won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) with top-1 error rate of 37.5%. Fine tuning to the Alex Net structure, mainly reducing the number of parameters involved in 2014, Szegedy et al., [10] introduced Goog Le Net that performed the best in their time with top-5 error rate of 6.67%. Karen Simonyan and Andrew Zisserman [11] of the University of Oxford also created a 19 layer CNN that strictly used 3x3 filters, a remarkable reduction compared to the two similar works discussed above which in turn means a reduction in the number of parameters achieved good performance because of the depth of the network.

The Microsoft Research Asia [12] brings new architecture with Res Net, a new 152 layer network architecture set new records in classification, detection, and localization through one incredible architecture that won ILSVRC 2015 with an incredible error rate of 3.6%. It followed the “divide & conquer” principle that improves performance regarding both speed and accuracy. The point here is the network learns the residual representation instead of the signal itself. Indeed, this work is after developing the “first to surpass the reported human-level performance model” [13] using rectifiers and advanced initialization method.

However, it is noteworthy to consider the work of Abramoff et al. [14] where they conclude that “automated detection of diabetic retinopathy using published algorithms cannot yet be recommended for clinical practice” after evaluating the performance of a system for automated detection of diabetic retinopathy in digital retinal photographs, built from published algorithms before, in a large, representative, screening population. Telemedicine-based diabetic retinopathy screening (DRS) in primary care settings can effectively detect sight-threatening retinopathy and significantly increase compliance with annual retinal exams, as stated by Cuadros, J. and G. Bresnick [15] on their work on Eye PACS. Eye PACS is a license-free Web-based DRS system designed to simplify the process of image capture, transmission, and review then the final decision still is to be done by certified professionals.
2. Methodology

The virtual hospital is built using unreal engine 4 (UE4). It includes buildings & the surrounding, the patient, health professionals, electronic equipment and furniture. There are few animations done including walking, taking eye picture, sleeping, and typing. All of the interaction to the virtual world is done via the keyboard. The player, we, can control the direction of movement of the patient, what to do, indeed with the help of widgets to know where he/she is and control the display units & other components of the project interacting with their blueprints. We have used makehuman to model humans and blender to bake different animations. Some of the models are designed using 3dsmax, and some animations are used from the official Mixamo website. Then Matlab is used to add the real-time image diagnosis.

Many existing medical image processing techniques depend on morphological feature representations to identify the local anatomical characteristics in which those representations mostly designed by human experts, which in turn requires intensive resource and the worst being problem-specific and hardly reusable for other image types.

A convolutional neural network is designed to utilize spatial and configuration data of images better. Structurally, CNNs have convolutional layers interspersed with pooling layers, followed by fully connected layers. A CNN takes advantage of weights sharing and subsampling.

To train models for image classification, we used the Kaggle dataset from the well-known home of data science, which is used for online competition in 2015 [15]. In this dataset, there are over 35000 labeled training and 53576 testing images, each with the different resolution taken with different modalities. Each image in this dataset is labeled with one of the five categories of diabetic retinopathy levels discussed below.

The 5 stages of DR severity that have symptoms as provided by the organizers of the competition:

0. non-pathological, no NPDR
1. Mild NPDR, microaneurysms (red blotches) which are the source of hard exudate (high contrast yellow spots) sometimes in circulating patterns
2. Moderate NPDR. “More than just microaneurysms,” perhaps cotton wool spots (fuzzy light blotches)
3. Severe NPDR: IRMA (shunt vessels), venous bleeding in 2+ quadrants, 20+ intra-retinal hemorrhages, no signs PDR
4. Neovascularization (often vessels with loops or very squiggly vessels), vitreous/pre-retinal hemorrhage, PDR

We should choose the best model among available pre-trained models and models designed from scratch for the image classification task. There are two categories in terms of design. The first being the model developed from scratch and the second is to fine-tune a few of the well-known pre-trained models with the same task, image classification, from Image Net. These two methods are implemented in two ways. The first way is classifying images in to the five classes given above. The other way is to divide the task into two stages. The primary stage is responsible for screening DR from Non-DR, whereas the secondary stage is to classify the DR cases into the remaining 4 stages of severity if the result of the primary stage is the second case.

2.1. Preprocessing

The purpose of image preprocessing before feeding the images to the designed model to minimize image variation by normalizing the original retinal image against the data set for subsequent viewing, processing or analysis. Variations mostly appear within the same image as well as between images and to get meaningful information from an image, and it is necessary to compensate for this variability though sometimes a little variation is essential in training, instead of adding extra noise as adversarial techniques use. Variations may arise due to differences in light diffusion, the presence of abnormalities, changes in fundus reflectivity and fundus thickness. Other variations include differences in cameras, illumination, acquisition angle, and retinal pigmentation. Both kinds of variations exist in the dataset used.

This stage is dataset-specific. Hence both the brightness and contrast enhancement techniques with some augmentations are applied. However, before that, since the original images are relatively large
(say, 3000x2000 pixels on average) and most contained a fairly significant black border, down-sampling to the designed model’s input size and to remove most of black borders cropping is applied. Hence, we resized each image to be 32,128,224 and 227 each being the same width and height and RGB (three channels). The following augmentation techniques are employed for the dataset.

1. Cropping the center with a certain probability
2. Brightness & Contrast adjustment (0.1,0.1)
3. Flipping images with 50% chance
4. Rotating images by x degrees, with x an integer in [0, 360]

These increases the size of the training set, which in turn is supposed to increase accuracy by decreasing over-fitting and also to keep the class imbalance though the result is the reverse that makes us stick on not using augmentation.

2.2. Network Architecture

Designing the architecture of a model involves choosing the types of layers and the way they are arranged and connected rather than just selecting the number of layers and hidden units with a few numbers of parameters which indeed can be done with the grid search option which means there are infinitely many ways in which one can design a network. Designing a good model is more of an art than science, usually involves a lot of trial and error that makes it depend on the personal experience of designing models and ideas come from good model architectures.

The main building blocks of a CNN are convolutional layers performed on the input data with the use of a filter to produce a feature map which is executed by sliding the filter over the input. At every location, matrix multiplication is performed and sums the result onto the feature map. The data depth relationship between two consecutive layers is governed by the equation (1) given below for each feature map which works for pooling layers too.

\[
N(l+1) = \frac{N(l) - NF(l) + 2P(l)}{S(l)} + 1
\]

Where:
1. \(N(l+1)\) is the data depth of the next layer
2. \(N(l)\) is the data depth of the current layer
3. \(NF\) is the filter size of the current layer
4. \(P\) is the padding applied
5. \(S\) is the stride used

So the hyper-parameters are the filter size (NF), stride(S), padding (P), and the number of filters. During both training and testing the input to our Conv Nets is a varying size RGB image as different models use different input sizes. The only preprocessing common for all the models is subtracting the mean RGB value, computed on the training set. Other processing techniques are also applied to check if worthy improving performance. The input image needs to pass through a stack of convolutional layers, where we use filters with a very small receptive field (i.e., 3 × 3 and 5x5). The convolution stride is 1 pixel except at a few places. Spatial pooling is carried out by max-pooling and avg-pooling layers, which follow some of the convolutional layers. Pooling is performed over a 2 by 2 pixel window, with stride two which decreases the size by half.

Figure 1 below shows one of the designed architectures as a two-phase diagnosis. The first one is feature extraction phase using convolutional, non-linearity, and pooling layers stuck as a block. Many models are designed using this architecture, just with different number of such blocks. These two layers, convolutional and pooling, serve as feature extractors. The last block is to predict the class of the image. The fully connected layer will serve as a classifier on top of the extracted features. Then the prediction layer (i.e., softmax), will give a probability for the output features based on the algorithm predicts.
Figure 1. Two-stage image classification method.

2.3. Divide and Conquer

Two models for the two stages of image classification task are proposed after trying many models. The first stage is to classify the image into two classes, No DR or there is DR while the second stage is to sort the image into one of the four types of DR if the result of the first stage is “There is DR”. Indeed the one stage classification is also applied for comparison. Comparing the two ways, using one stage and two stages, we found that the later has better accuracy that leads us to implement the second way. We can see from the dataset that large portion of it is Non-DR case though it doesn’t mean that we should rely on this premise. The point is to reduce the resources wasted for this large percentage of the cases if we don’t implement such automated screening. So if we can have a model well enough for the first stage, the information we get is paramount important from the purpose of this work’s point of view. Hence, if the result of the first stage is “No DR,” we don’t need to worry about it though we can warn the patient depending on the level of accuracy to do the screening regularly. However, if the result turns the other way, further classification is required using the second model.

3. Results

After many trials of different models for image diagnosis part, below are some of the top performers to be compared. Apart from the two ways, we have also tried fine-tuning for feature extraction and SVM as a classifier in which the result was not worthy attractive for further improvement.

| Model No. | Model Name/type               | Accuracy (one-stage diagnosis) |
|-----------|-------------------------------|--------------------------------|
| 1         | Batch normalized simple network | 0.77890                        |
| 2         | Deeper network with a smaller filter size | 0.69000                  |
| 3         | Series/Dag networks           | 0.74210                        |
| 4         | Fine-tuning VGG16             | 0.73900                        |
| 5         | Fine-tuning AlexNet           | 0.72790                        |
| 6         | Fine-tuning GoogleNet         | 0.73789                        |
| 7         | Fine-tuning Resnet50          | 0.73596                        |

| Model No. | Model Name/type               | Accuracy  |
|-----------|-------------------------------|-----------|
|           |                                | Stage-1   | Stage-2   |
| 1         | Batch normalized simple network | 0.8712    | 0.6447    |
| 2         | Deeper network with a smaller filter size | 0.5429    | 0.6111    |
| 3         | Series/Dag networks           | 0.7368    | 0.5598    |
| 4         | Fine-tuning VGG16             | 0.7379    | 0.5598    |
| 5         | Fine-tuning AlexNet           | 0.7241    | 0.5431    |
| 6         | Fine-tuning GoogleNet         | 0.7379    | 0.5598    |
We can see the difference in accuracy of the two ways from the two tables for all the models. So the second way with the first model from the table 2 above is used for the image classification task. The first stage is to classify the image into two classes (DR and No DR) and then based on this result the second stage is to classify into one of the four types of DR if the result of the primary stage is a DR case. Having chosen how to deal with the task, now let’s see the effect of further improvements on this model with extra image processing as presented below.

Table 3. The improvements in accuracy by applying further processing with the selected model.

| Model No. | Improvements applied                        | Accuracy   |
|-----------|--------------------------------------------|------------|
|           |                                            | Stage-1    | Stage-2    |
| 1         | With only data normalization               | 0.8712     | 0.6447     |
| 2         | Brightness, contrast, and flipping         | 0.7377     | 0.5883     |
| 3         | Augmentation with flipping                 | 0.7604     | 0.6454     |
| 4         | Brightness and contrast enhancement        | 0.7686     | 0.7189     |

Figure 2. Instance of the training progress (objective and top-1 error) & prediction probabilities of models designed for stage-1.

Finally, a standalone application which asks the user for the location of the images with the help of a dialog box as shown in the figure below is created. With the help of the Matlab Compiler Runtime (MCR), it can run on a standard Windows PC without the need of Matlab installation. It is used to save the image and display some useful information for the patient in the virtual hospital when the patient presses a key on the keyboard as depicted on figure three below.
Figure 3. This figure shows some of the interfaces of our work. (a) The Matlab dialog box to get image path information; (b) The patient interaction in UE4 to see the diagnosis result; (c) The result of image diagnosis displayed for the user.

4. Conclusion
From this experience, we can conclude without any doubt that deep learning is a potent tool for medical imaging applications. The representative power of neural networks is impressive, and they have been specialized for use in medical imaging turning tasks that seem very hard, or even impossible, into reasonable ones.

As we can see the results from section three above, the designed models perform better compared to the well-known pre-trained models on the same task, image classification, taking the accuracy as main comparison parameter. It is evident from the three tables that fine-tuning looks saturated in terms of accuracy improvements. Therefore our models are finally the best candidate for, at least the first stage DR screening, which can be considered as a first step towards practical applications in health care units.

From figure two above, we can see that the top-1 error of the designed model is 0.267 for stage-1. The best state of the art model from ILSVRC image classification, Resnet-152 the winner of the competition, has less than 0.2 top-1 error rate although the dataset is totally different. So our model has very close result, for stage-1, with significant accuracy improvement. The techniques applied to further improve the accuracy, table 3, achieved the goal regarding accuracy although the top-1 error rate has shown no further improvements.

All in all, this is the first application in its kind, real-time image diagnosis used in the virtual hospital, to serve as part of the virtual environment platform for whatever purpose. The critical contribution of this work is to lay the foundation for a general and extensible framework which can potentially include tele-immersive technologies and applications in the future to fully automate the process of classification of diabetic retinopathy by adding a level of intelligence on works like an adaptable telemedicine system for diabetic retinopathy screening [15] using the success of CNN on image classification.

5. Acknowledgments
This work was supported by Northwest Polytechnic University Master Degree Student Creative Innovation Seed Fund Project, grant number ZZ2018132. We also gratefully acknowledge the support of Northwestern Polytechnic University Internet Service Center for their help in downloading the dataset used in this research.

6. References
[1] Robb, R.A. Medical imaging and virtual reality: A personal perspective. Virtual Real. 2008, 12, 235-257.
[2] Govea, E.; Medellín-Castillo, H.I. Design and development of virtual reality environments for biomedical and engineering applications. 2015, V014T006A007.
[3] Govea-Valladares, E.H.; Medellín-Castillo, H.I.; Ballesteros, J.; Rodríguez-Florido, M.A. On the development of virtual reality scenarios for computer-assisted biomedical applications. Journal of Healthcare Engineering 2018, 2018, 13
[4] Filippi, S.; Motyl, B.; Bandera, C. Analysis of existing methods for 3d modelling of femurs starting from two orthogonal images and development of a script for a commercial software package. Computer methods and programs in biomedicine 2008, 89, 76-82.
[5] Petrasek, D.; Barr, A.; Palem, K.V. The virtual hospital: The emergence of telemedicine. In Proceedings of the 2010 international conference on Compilers, architectures and synthesis for embedded systems, ACM: Scottsdale, Arizona, USA, 2010; pp 53-54.
[6] Singh, R.; Singh, A. K. Transcending from virtual reality into tele-immersive technologies and applications: A perspective. Ubiquity 2008, 2008, 3-3.
[7] Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence 2013, 35, 1798-1828.
[8] Shen, D.; Wu, G.; Suk, H.I. Deep learning in medical image analysis. Annual review of biomedical engineering 2017, 19, 221-248.
[9] Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, Curran Associates Inc.: Lake Tahoe, Nevada, 2012; pp 1097-1105.
[10] Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. CoRR 2014, abs/1409.1556.
[11] He, K.; Zhang, X.; Ren, S.; Sun, J. In Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015 IEEE International Conference on Computer Vision (ICCV), 7-13 Dec. 2015, 2015; pp 1026-1034.
[12] He, K.; Zhang, X.; Ren, S.; Sun, J. In Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015 IEEE International Conference on Computer Vision (ICCV), 7-13 Dec. 2015, 2015; pp 1026-1034.
[13] He, K.; Zhang, X.; Ren, S.; Sun, J. In Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015 IEEE International Conference on Computer Vision (ICCV), 7-13 Dec. 2015, 2015; pp 1026-1034.
[14] Abramoff, M.D.; Niemeijer, M.; Suttorp-Schulten, M.S.; Viergever, M.A.; Russell, S.R.; van Ginneken, B. Evaluation of a system for automatic detection of diabetic retinopathy from color fundus photographs in a large population of patients with diabetes. Diabetes care 2008, 31, 193-198.
[15] Cuadros, J.; Bresnick, G. Eyepacs: An adaptable telemedicine system for diabetic retinopathy screening. Journal of diabetes science and technology 2009, 3, 509-516.