Báo cáo đồ án 2:

Nucleus Segmentation using U-Net

Giảng viên hướng dẫn : TS. Phạm Văn Trường

Sinh viên thực hiện : Bùi Duy Cường

Lớp : KSTN-ĐKTĐ k61

MSSV : 20160525

Mục lục

[A. Giới thiệu 3](#_Toc63767245)

[**B.** **Mô hình** 4](#_Toc63767246)

[*I.* *Tổng quan kiến trúc U-net* 5](#_Toc63767247)

[**C.** **Thực nghiệm.** 10](#_Toc63767248)

[*1.* *Môi trường:* 10](#_Toc63767249)

[*2.* *Tiền xử lý.* 10](#_Toc63767250)

[*3.* *Training* 13](#_Toc63767251)

[**D.** **Kết luận** 19](#_Toc63767252)

[**E.** **Danh mục tài liệu tham khảo:** 20](#_Toc63767253)

1. Giới thiệu

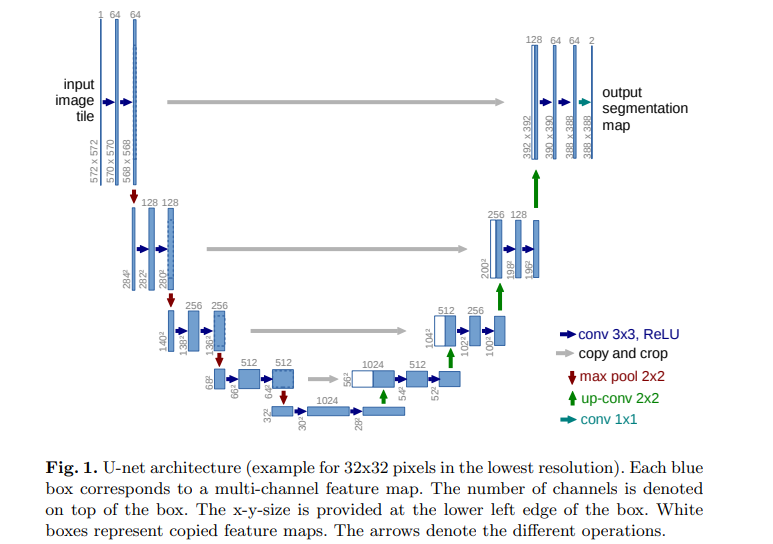
*Ứng dụng Deeplearning trong phân tích ảnh y sinh*

Kể từ khi ra đời thiết bị hình ảnh y tế kỹ thuật số, việc áp dụng các kỹ thuật xử lý hình ảnh trong phân tích hình ảnh y tế đã dành được sự chú ý đáng kể. Các nhà nghiên cứu đa ngành đã làm việc siêng năng trong nhiều thập kỷ để phát triển hệ thống chẩn đoán tự động và cho đến ngày nay nó là một trong những lĩnh vực nghiên cứu tích cực nhất. Nhiệm vụ của một công cụ phân tích hình ảnh y tế có sự hỗ trợ của máy tính gồm hai phần: phân đoạn(segmentation) và chẩn đoán(diagnosis). Trong bài toán phân đoạn chung, mục tiêu là phân vùng một hình ảnh thành một tập hợp các vùng không chồng chéo, cho phép đồng nhất các pixel được nhóm lại với nhau. Tuy nhiên, trong bối cảnh y tế, sự quan tâm thường chỉ nằm ở việc phân biệt một số khu vực đặc biệt của hình ảnh, như các vùng khối u, các cơ quan, … Điều này cho phép các bác sĩ chỉ phân tích những phần quan trọng của các hình ảnh y tế. Hơn nữa, các hình ảnh được phân đoạn thường được sử dụng để tính toán các tính năng khác nhau có thể được tận dụng trong chẩn đoán. Do đó, phân đoạn hình ảnh là vô cùng quan trọng và ứng dụng to lớn trong lĩnh vực Kỹ thuật Y sinh. Do ý nghĩa sâu sắc của việc phân đoạn hình ảnh y tế và sự phức tạp liên quan đến việc thực hiện điều đó theo cách thủcông, một số lượng lớn các phương pháp phân đoạn hình ảnh y tế tự động đã được phát triển, chủ yếu tập trung vào hình ảnh của các phương thức cụ thể. Trong những ngày đầu, các phương pháp tiếp cận dựa trên quy tắc đơn giản được thực hiện. Tuy nhiên, những phương pháp đó không duy trì được hiệu quả khi được thử nghiệm trên nhiều dữ liệu khác nhau. Do đó, các thuật toán thích ứng hơn đã được phá ttriển dựa trên các mô hình dạng hình học với các công cụ của máy tính mềm và hệ thống mờ. Tuy nhiên, những phương pháp này bị ảnh hưởng bởi thành kiến của con người và không thể xử lý sự khác biệt trong dữ liệu trong thế giới thực. Những tiến bộ gần đây trong Deeplearning cho thấy rất nhiều hứa hẹn đối với việc giải quyết các vấn đề trên. Về vấn đề này, Convolutional Neural Networks - CNN là sự bổ sung đột phá nhất, và phổ biến nhất trong lĩnh vực Thị giác máy tính. CNN chịu trách nhiệm về những tiến bộ vượt bậc trongcác nhiệm vụ như phân loại đối tượng (object classification), bản địa hóa đối tượng (object localization),… và những cải tiến liên tục đối với kiến trúc CNN đang mang lại những tiến bộ triệt để hơn nữa.

Mặc dù tạo ra một bước đột phá trong các nhiệm vụ thị giác máy tính, một nhược điểm lớn của kiến trúc CNN là chúng đòi hỏi khối lượng lớn dữ liệu để training. Thật không may, trong bối cảnh của y tế, không chỉ việc thu thập hình ảnh là tốn kém và phức tạp, độ chính xác của chúng thậm chí còn làm tăng thêm sự phức tạp. Tuy nhiên, CNN đã cho thấy những hứa hẹn lớn trong việc phân khúc hình ảnh y tế trong những năm gần đây, và hầu hết các credit được chuyển đến U-Net. Cấu trúc của U-Net khá giống với SegNet, bao gồm một encoder và một mạng decoder. Hơn nữa, các lớp tương ứng  
của mạng encoder và decoder được kết nối với nhau bằng skip connection trước khi gộp và sau đó là hoạt động giải mã tương ứng. U-Net đã và đang cho thấy tiềm năng ấn tượng trong việc phân đoạn các hình ảnh y tế, ngay cả với lượng dữ liệu đào tạo được gắn nhãn khan hiếm. Các mô hình tương tự U-Net đã được sử dụng thành công trong phân đoạn các hình ảnh y sinh của cấu trúc tế bào thần kinh, gan, tổn thương da, thận, tuyến tiền liệt,...

Ở trong báo cáo này em sẽ trình bày về ứng dụng của U-net trong segmentation các tế bào hồng cầu. Từ đó có thể đưa ra được các dự đoán về tình trạng cơ thể, số lượng tế bào máu hay các bệnh liên quan, ….

1. **Mô hình**
2. *Tổng quan kiến trúc U-net*



Tương tự với FCN và SegNet , U-Net sử dụng một mạng lưới các lớp phức hợp hoàn toàn để thực hiện nhiệm vụ semantic segmentationn. Kiến trúc mạng là đối xứng, bao gồm encoder trích xuất các spatial features từ hình ảnh và decoder xây dựng bản đồ phân đoạn (segmentation map) từ các features được mã hóa. Encoder tuân theo sự hình thành điển hình của convolutional network. Nó liên quan đến một chuỗi hai phép toán tích chập 3 × 3 (convolution operations) , tiếp theo là (max-pooling operation) với window size là 2 × 2 và bước tiến(stride)là 2. Chuỗi này được lặp lại bốn lần và sau mỗi lần lấy mẫu giảm, số lượng bộ lọc trong các lớp chập (convolutional layers) được nhân đôi. Cuối cùng, một tiến trình của hai phép toán tích chập 3 × 3 (convolution operations) kết nối Encoder với Decoder. Mặt khác, Decoder trước tiên lấy feature map bằng cách sử dụng phép toán tích chập chuyển vị 2 × 2 (transposed convolution operation), giảm một nửa số kênh đối tượng. Sau đó, một chuỗi hai phép tính chập 3 × 3 (convolution operations) được thực hiện lại Tương tự như Encoder, phép lấy mẫu và hai phép tích chập này được lặp lại bốn lần, làm giảm một nửa số bộ lọc ở mỗi giai đoạn. Cuối cùng, phép toán tích chập 1 × 1 được thực hiện để tạo ra bản đồ phân đoạn (segmentation map) cuối  
cùng. Tất cả các lớp convolutional layers trong kiến trúc này, ngoại trừ lớp cuối cùng, sử dụng chức năng kích hoạt ReLU (Rectified Linear Unit); lớp tích chập cuối cùng sử dụng active function Sigmoid. Trong cả bốn cấp, đầu ra của lớp tích chập, trước khi hoạt động gộp của Encoder xảy ra, được chuyển đến  
Decoder. Sau đó, feature map này được nối với đầu ra của thao tác lấy mẫu ngược (upsampling operation) và được truyền đến các lớp kế tiếp. Các skip connections này cho phép mạng truy xuất thông tin không gian (spatial information) bị mất do hoạt động gộp Kiến trúc mạng được minh họa trong Hình 1.

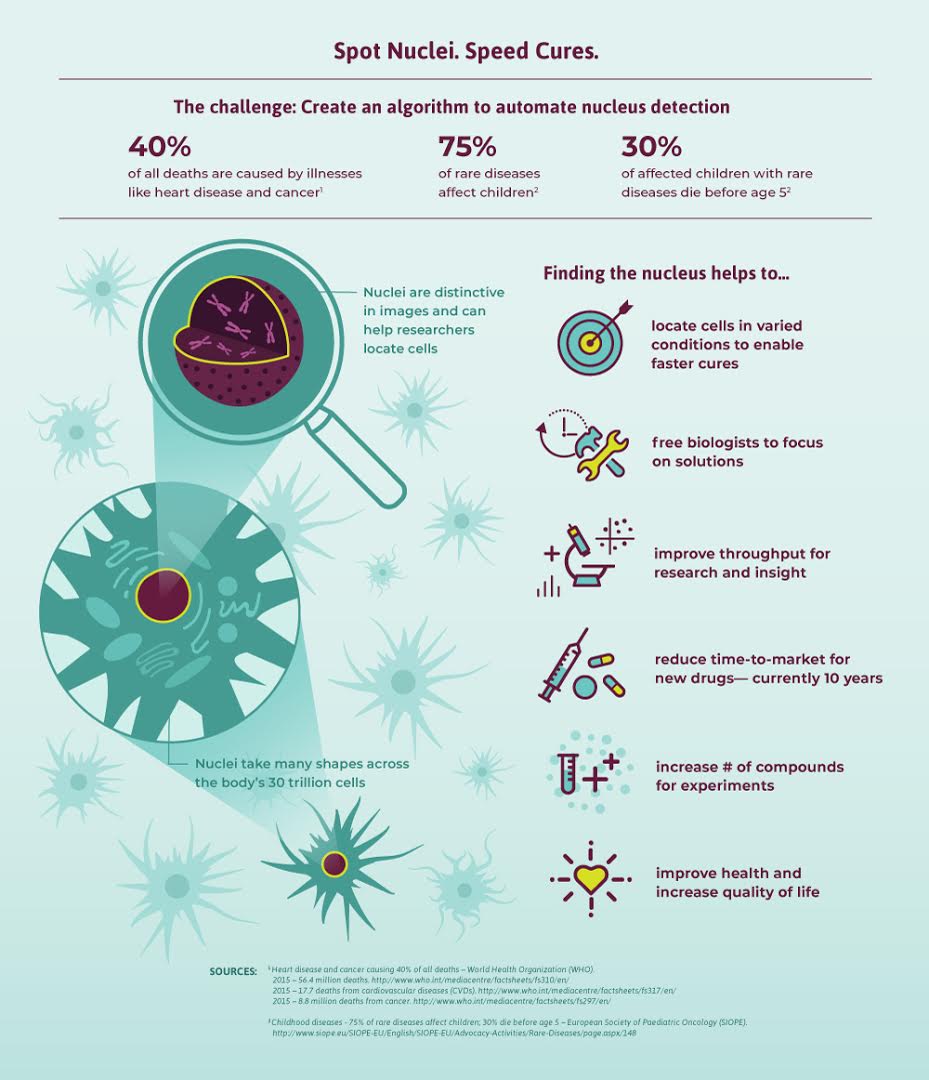
Sau đó, kiến trúc U-Net được mở rộng thông qua một vài sửa đổi đối với 3D UNet để phân đoạn theo thể tích (volumetric segmentation). Đặc biệt, các phép toán tích chập hai chiều (two-dimensional convolution), max pooling, transposed convolution operations đã được thay thế bằng các phép toán trong không gian ba chiều của chúng. Tuy nhiên, để hạn chế số lượng tham số, độ sâu của mạng đã được giảm một phần. Hơn nữa, cùng với việc sử dụng batch normalization, số lượng bộ lọc đã được tăng gấp đôi trước các lớp gộp để tránh tắc nghẽn. U-Net cổ điển không sử dụng batch normalization. Tuy nhiên, khi thử nghiệm với nó sau đó, kết quả là việc chuẩn hóa hàng loạt thậm chí đôi khi có thể làm ảnh hưởng đến hiệu suất.

1. *Dataset.*

Dataset chúng ta sử dụng trong bài này là tập Data Science Bowl 2018 do công ty Booz Allen Hamilton, một công ty công nghệ thông tin của Mỹ, có trụ sở tại McLean, Virginia, Washington DC đưa lên để tổ chức cuộc thi Nucleus Segmentation với giải thưởng lên tới 100,000$. Cuộc thi thu hút hơn 3000 đội tham gia. Các đội sẽ tạo ra một mô hình máy tính có thể xác định một loạt các hạt nhân trong các điều kiện khác nhau. Bằng cách quan sát các mẫu, đặt câu hỏi và xây dựng một mô hình, người tham gia sẽ có cơ hội đẩy công nghệ hiện đại đi xa hơn.

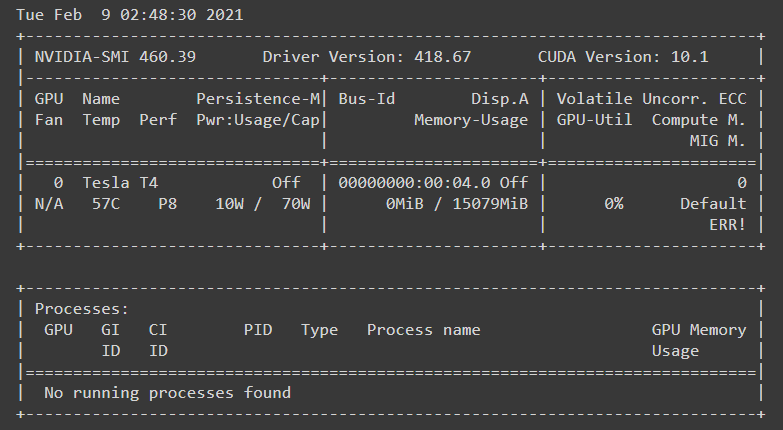
Mỗi ảnh trong tập Data Science Bowl 2018 được biểu thị bởi 1 ImageID. Mỗi ImageID là tên 1 thư mục, bên trong sẽ bao gồm 2 subFolder:

* Images: Chứa file ảnh
* Masks: Chứa mask đã segment của từng tế bào. Folder này chỉ được chứa ở trong training set. Các mask không được trùng lặp lên nhau (không có pixel nào thuộc về cả 2 mask).
* Tập dữ liệu thứ hai sẽ chứa các hình ảnh từ các điều kiện thử nghiệm không nhìn thấy. Để ngăn chặn việc ghi nhãn bằng tay, nó cũng sẽ chứa những hình ảnh bị bỏ qua khi cho điểm. Chỉ số được sử dụng để tính điểm cho cuộc thi này yêu cầu bài nộp phải ở định dạng được mã hóa trong thời gian dài.



1. **Thực nghiệm.**
2. *Môi trường:*

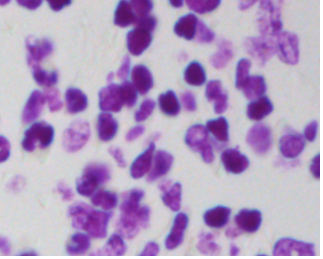
Colab, GPU phiên bản như sau



1. *Tiền xử lý.*

Vì dataset training gồm 2 phần, là images và masks. Trong khi images là thư mục chứa các thư mục con ảnh thì masks chứa các ảnh mask của từng tế bào. Vì thế để có thể training ta cần đưa tất cả các mask của mỗi image về cùng một ảnh. Sau đó để tránh tràn ram nên ta dùng một bộ data\_generator đưa từ từ dữ liệu training vào model theo batch\_size.

Image và Masks trước khi tiền xử lý:



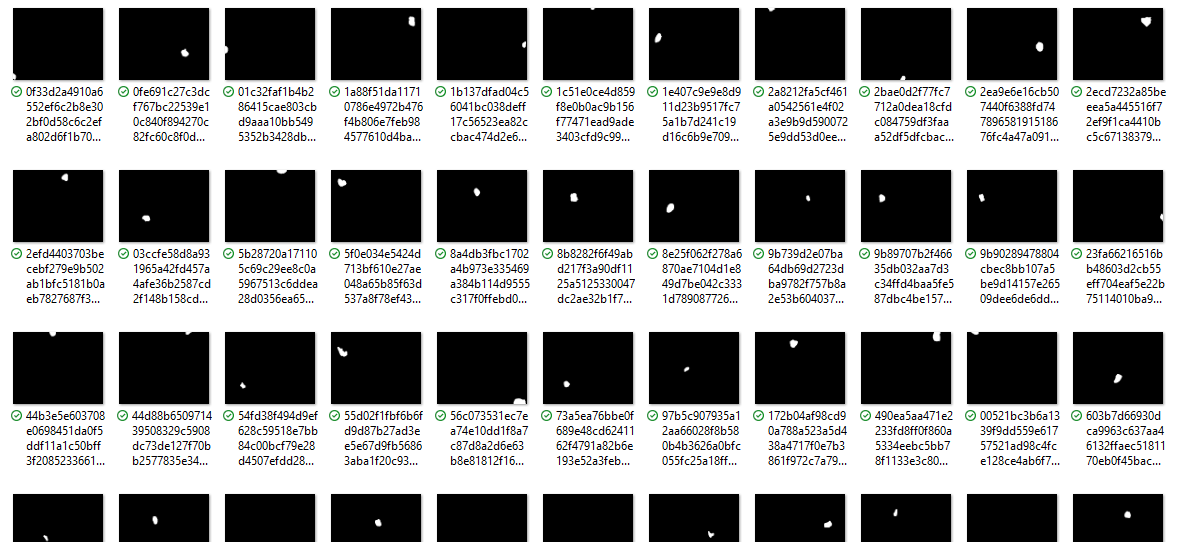
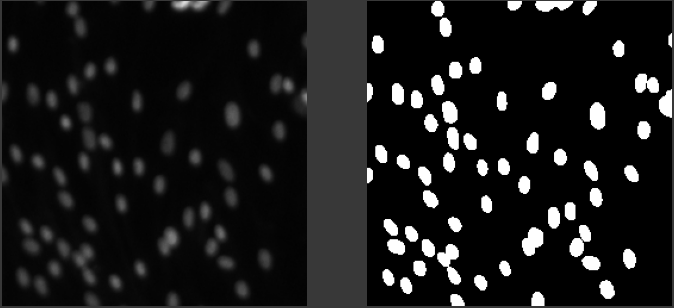
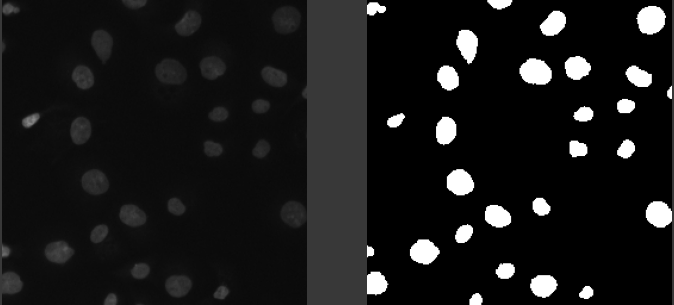
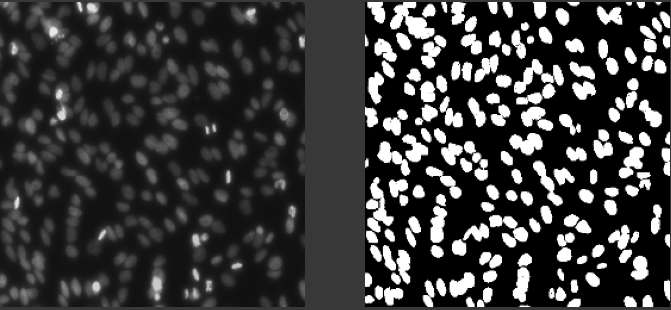


Image và Mask sau khi tiền xử lý:



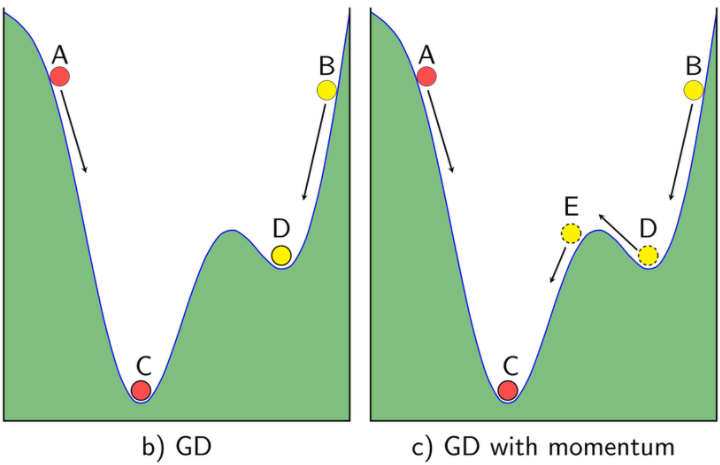




1. *Training*
2. *Hàm tối ưu: Dùng để tính toán nghiệm hội tụ* .

* Gradient Descent / Gradient Descent with Momentum

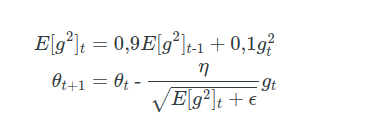
Công thức:

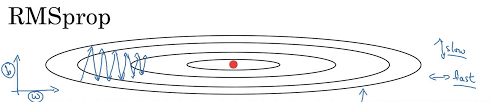
***xnew = xold -(gama.v + learningrate.gradient)***

**Ưu điểm :**  
 Thuật toán gradient descent cơ bản, dễ hiểu. Thuật toán đã giải quyết được vấn đề tối ưu model neural network bằng cách cập nhật trọng số sau mỗi vòng lặp.  
**Nhược điểm :**  
 Vì đơn giản nên thuật toán Gradient Descent còn nhiều hạn chế như phụ thuộc vào nghiệm khởi tạo ban đầu và learning rate.

* *RMSprop*

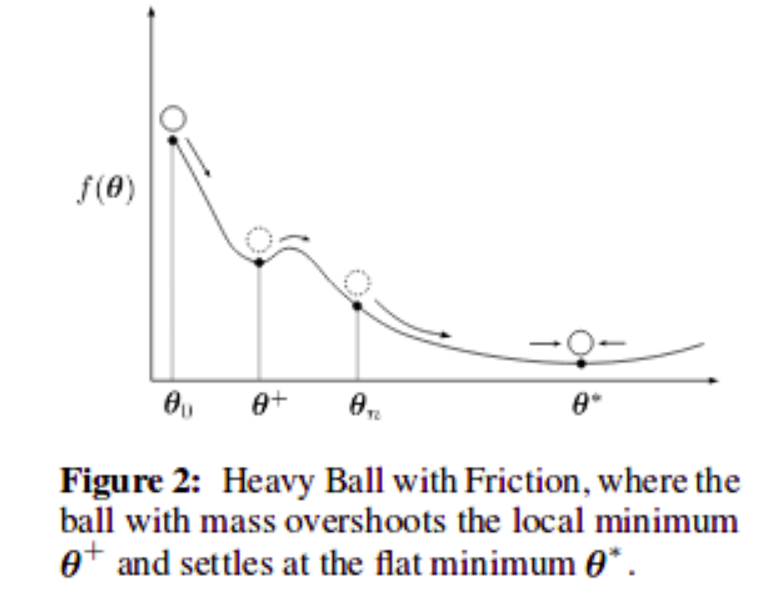
Công thức:



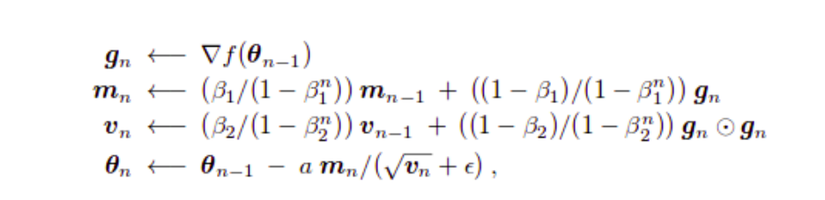


**Ưu điểm :**  
 Ưu điểm rõ nhất của RMSprop là giải quyết được vấn đề tốc độ học giảm dần ( vấn đề tốc độ học giảm dần theo thời gian sẽ khiến việc training chậm dần, có thể dẫn tới bị đóng băng )  
**Nhược điểm :**  
 Thuật toán RMSprop có thể cho kết quả nghiệm chỉ là local minimum chứ không đạt được global minimum như Momentum. Vì vậy người ta sẽ kết hợp cả 2 thuật toán Momentum với RMSprop cho ra 1 thuật toán tối ưu Adam.

* *Adam*



Công thức:

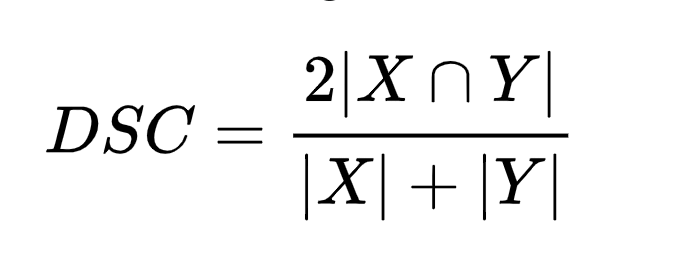


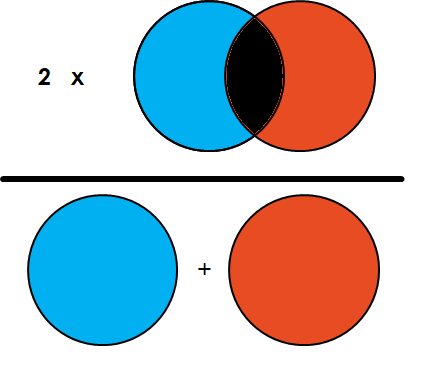
***Ta thấy rằng thuật toán tối ưu Adam hoạt động khá tốt. Ở trong bài này ta sử dụng thuật toán tối ưu Adam, giúp loại trừ các nhược điểm của các thuật toán tối ưu trên mà vẫn giữ lại được các ưu điểm của chúng.***

1. *Loss Function:*

Dice\_coef: Bằng 2 lần diện tích phần giao nhau của mask thực tế và mask dự đoán chia cho tổng của chúng

Công thức:



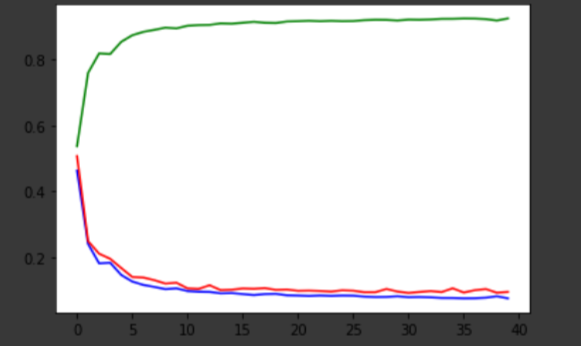


Model của chúng ta sẽ dự đoán chính xác nhất khi 2 mask thực tế và mask dự đoán giống hệt nhau. Tức là khi đó dice\_coef = 1.

Tuy nhiên, trên thực tế việc sử dụng optimizer trong Machine learning là tối thiểu hóa hàm Loss nên khi đó ta sử dụng hàm:

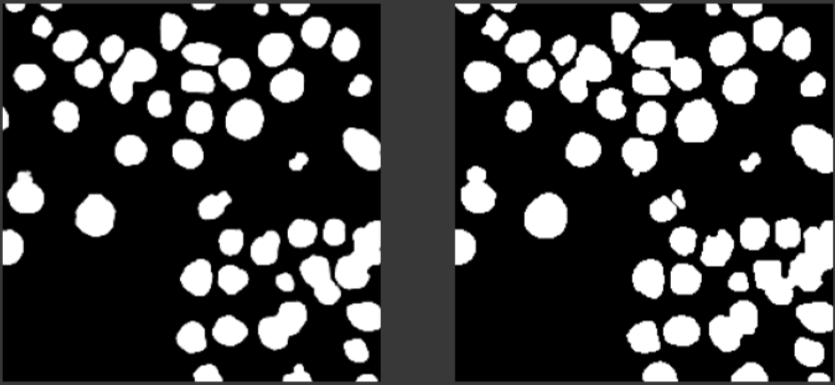
**dice\_coef\_loss = 1 – dice\_coef**

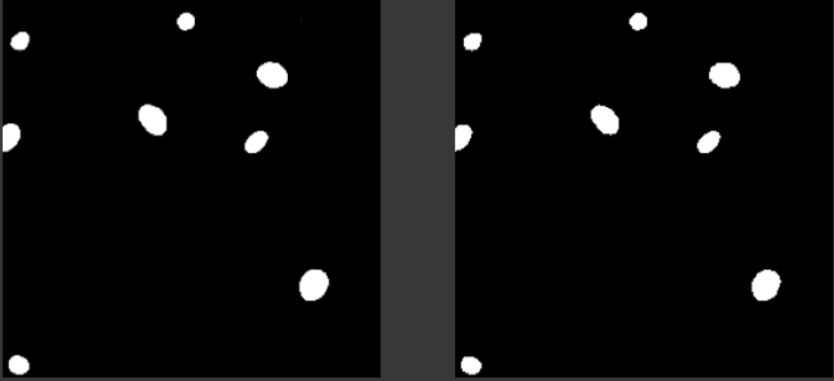
1. *Kết quả:*

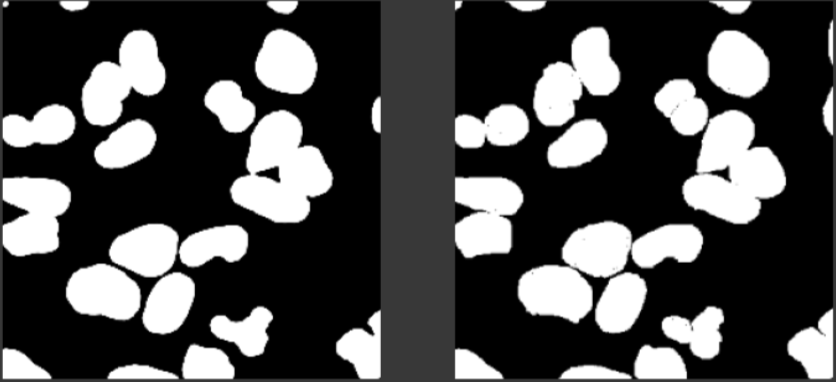
****

**Đồ thị Loss Function**

**Ground Truth và Predict:**

****

****

****

**Ta có thể thấy Predict khá gần với Ground Truth với độ chính xác xấp xỉ 92% sau 40 epochs.**

1. **Kết luận**

Ứng dụng của Machine Learning trong phân tích ảnh y sinh là vô cùng quan trọng, giúp cho công việc của người bác sỹ trở lên đơn giản hơn rất nhiều, với độ chính xác cao và chi phí vận hành thấp. Em tin rằng việc kết hợp mô hình của mình với các giai đoạn xử lý sau thích hợp sẽ cải thiện hiệu suất mô hình của hơn nữa và cho phép phát triển các phương pháp phân đoạn tốt hơn cho các ứng dụng đa dạng. Trong khuôn khổ của môn học, hiện tại em mới chỉ dừng công việc ở đưa ra các masks cho hình ảnh. Em mong rằng trong tương lai có thể phát triển mô hình hơn để có thể đưa ra kết quả chính xác hơn, có nhiều ứng dụng hơn để giúp ích cho lĩnh vực phân tích hình ảnh y sinh.

Em xin chân thành cám ơn.

1. **Danh mục tài liệu tham khảo:**
2. [**https://arxiv.org/pdf/1505.04597.pdf**](https://arxiv.org/pdf/1505.04597.pdf)
3. [**https://www.kaggle.com/c/data-science-bowl-2018**](https://www.kaggle.com/c/data-science-bowl-2018)
4. [**https://towardsdatascience.com/metrics-to-evaluate-your-semantic-segmentation-model-6bcb99639aa2**](https://towardsdatascience.com/metrics-to-evaluate-your-semantic-segmentation-model-6bcb99639aa2)
5. [**https://keras.io/api/optimizers/**](https://keras.io/api/optimizers/)
6. [**https://www.nature.com/articles/s41592-019-0612-7**](https://www.nature.com/articles/s41592-019-0612-7)