Deep Learning Analysis in Dermoscopy Images

Fatih Ergin

Galatasaray University

Department of Computer Engineering

Outline

- Motivation
- Image Segmentation Architectures
- Methodology
- Results
- Discussions and Future Works
- Conclusion

Outline

- Motivation
- Image Segmentation Architectures
- Methodology
- Results
- Discussions and Future Works
- Conclusion

Motivation

- Background of the Problem
- Aim and Scope
- Challenges
- Why Deep Learning now?

Motivation

- Background of the Problem
- Aim and Scope
- Challenges
- Why Deep Learning now?

Motivation

- Background of the Problem
  - Skin cancer: 100.000+ estimated new case 2020

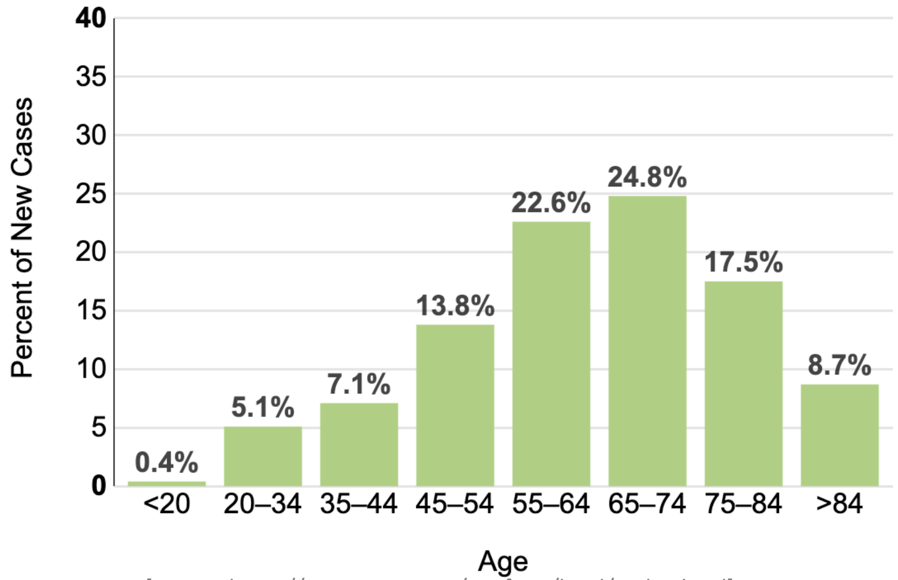
	Common Types of Cancer	Estimated New Cases 2020	Estimated Deaths 2020
1.	Breast Cancer (Female)	276,480	42,170
2.	Lung and Bronchus Cancer	228,820	135,720
3.	Prostate Cancer	191,930	33,330
4.	Colorectal Cancer	147,950	53,200
5.	**Melanoma of the Skin**	**100,350**	**6,850**
6.	Bladder Cancer	81,400	17,980
7.	Non-Hodgkin Lymphoma	77,240	19,940
8.	Kidney and Renal Pelvis Cancer	73,750	14,830
9.	Uterine Cancer	65,620	12,590
10.	Leukemia	60,530	23,100

Melanoma of the skin represents 5.6% of all new cancer cases in the U.S.

[source: https://seer.cancer.gov/statfacts/html/melan.html]

Motivation

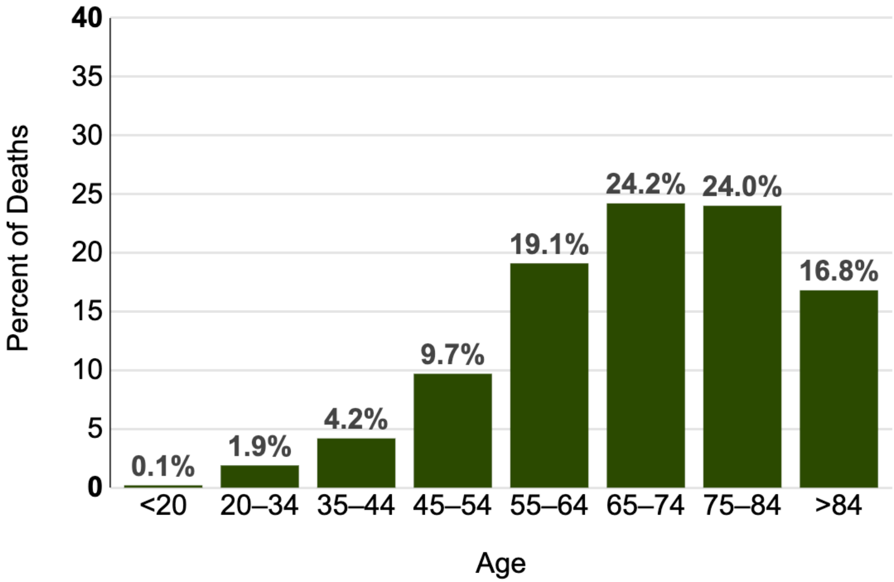
- Background of the Problem
  - Skin cancer: 100.000+ estimated new case 2020
  - Early detection matters



[source: https://seer.cancer.gov/statfacts/html/melan.html]

Motivation

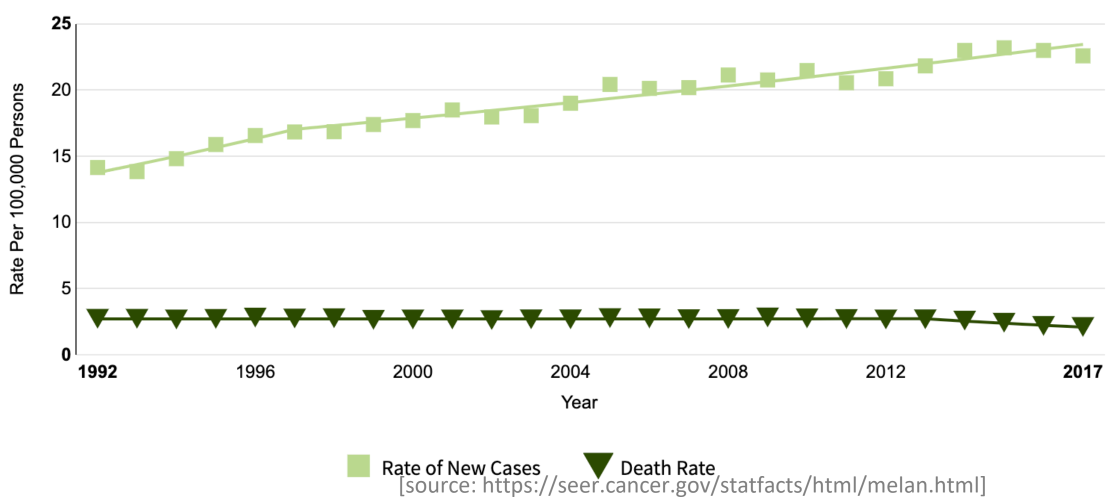
- Background of the Problem
  - Skin cancer: 100.000+ estimated new case 2020
  - Early detection matters
  - Can be deadly for all ages



[source: https://seer.cancer.gov/statfacts/html/melan.html]

Motivation

- Background of the Problem
  - Skin cancer: 100.000+ estimated new case 2020
  - Early detection matters
  - Can be deadly for all ages
  - Increasing mortality



[source: https://seer.cancer.gov/statfacts/html/melan.html]

Motivation

- Background of the Problem
- Aim and Scope
- Challenges
- Why Deep Learning now?

Motivation

- Aim and Scope
  - Aim
    - Investigate the success of DL models on pattern recognition
    - Measure the robustness of the proposed models
    - As a result: Help physicians to detect melanoma

Motivation

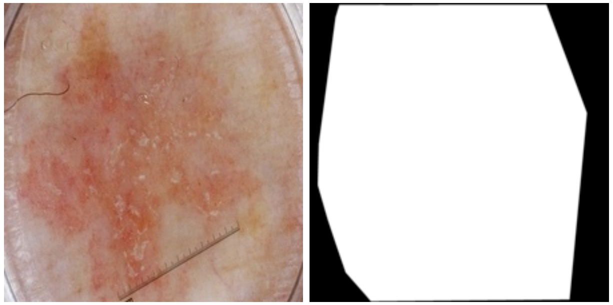
- Aim and Scope
  - Aim
    - Investigate the success of DL models on pattern recognition
    - Measure the robustness of the proposed models
    - As a result: Help physicians to detect melanoma
  - Scope
    - Evaluate RGB dermoscopy images in various dimensions
    - Adapt state of the art models to skin lesion problem
      - MultiResUNet, SegAN
    - Compare the results with a well-known benchmark: U-Net
    - Test the accuracy in non trained noisy data
    - Prove if SegAN is more resistant to noise

Motivation

- Background of the Problem
- Aim and Scope
- Challenges
- Why Deep Learning now?

Motivation

- Challenges
  - For physicians
    - Artifacts
      - Moles, freckles, hair, shading, noise
    - Irregular, fuzzy lesion border
    - Low contrast between lesion and surrounding skin
    - Multiple skin lesion



Motivation

- Challenges
  - For physicians
    - Artifacts
      - Moles, freckles, hair, shading, noise
    - Irregular, fuzzy lesion border
    - Low contrast between lesion and surrounding skin
    - Multiple skin lesion
  - For data scientists
    - Data collection & preparation
    - Implementation of neural networks
    - Hardware requirements

Motivation

- Background of the Problem
- Aim and Scope
- Challenges
- Why Deep Learning now?

Motivation

- Why Deep Learning now?
  - Large datasets
    - MNIST, CIFAR-10, COIL, ImageNet, MS COCO, Flickr,

Motivation

- Why Deep Learning now?
  - Large datasets
    - MNIST, CIFAR-10, COIL, ImageNet, MS COCO, Flickr,
  - Increased computational power
    - GPU, TPU

Motivation

- Why Deep Learning now?
  - Large datasets
    - MNIST, CIFAR-10, COIL, ImageNet, MS COCO, Flickr,
  - Increased computational power
    - GPU, TPU
  - Efficient, optimized algorithms; networks

Motivation

- Why Deep Learning now?
  - Large datasets
    - MNIST, CIFAR-10, COIL, ImageNet, MS COCO, Flickr,
  - Increased computational power
    - GPU, TPU
  - Efficient, optimized algorithms; networks
  - Easy to use frameworks
    - Tensorflow, Keras, PyTorch, Caffe

Outline

- Motivation
- Image Segmentation Architectures
- Methodology
- Results
- Discussions and Future Works
- Conclusion

Image Segmentation Architectures

- Image Segmentation
- Classical Methods
- Brief history of DL
- Common Image Segmentation Architectures

Image Segmentation Architectures

- Image Segmentation
- Classical Methods
- Brief history of DL
- Common Image Segmentation Architectures

Image Segmentation Architectures

- Image Segmentation
  - Partitioning an image into distinctive subset

a) b) c)

Image Segmentation Architectures

- Image Segmentation
- Classical Methods
- Brief history of DL
- Common Image Segmentation Architectures

Image Segmentation Architectures

- Classical Methods
  - Similarity detection based approach (Region)
    - Thresholding, region growing, region splitting and merging, clustering, watershed

Image Segmentation Architectures

- Classical Methods
  - Similarity detection based approach (Region)
    - Thresholding, region growing, region splitting and merging, clustering, watershed
  - Discontinuity detection based approaches (Edge)
    - Edge detection
      - Sobel, Canny, Laplacian

Image Segmentation Architectures

- Image Segmentation
- Classical Methods
- Brief history of DL
- Common Image Segmentation Architectures

Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Artificial Neural Networks
    - Convolutional Neural Networks
    - Fully Convolutional Networks
  - Is it a blackbox?

Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Artificial Neural Network
      - Ivakhnenko and Lapa in 1965
      - Remarkable results between 2009 – 2012
        - pattern recognition and machine learning*

*[source: Schmidhuber, J. (2015). "Deep Learning in Neural Networks: An Overview". *Neural Networks*. **61**: 85–117.]

Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Artificial Neural Network
      - Ivakhnenko and Lapa in 1965
      - Remarkable results between 2009 – 2012
        - pattern recognition and machine learning

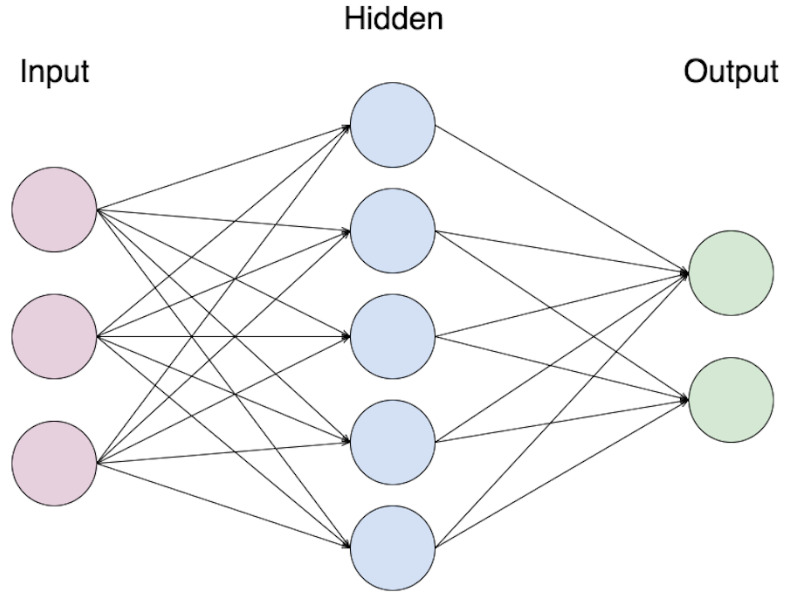
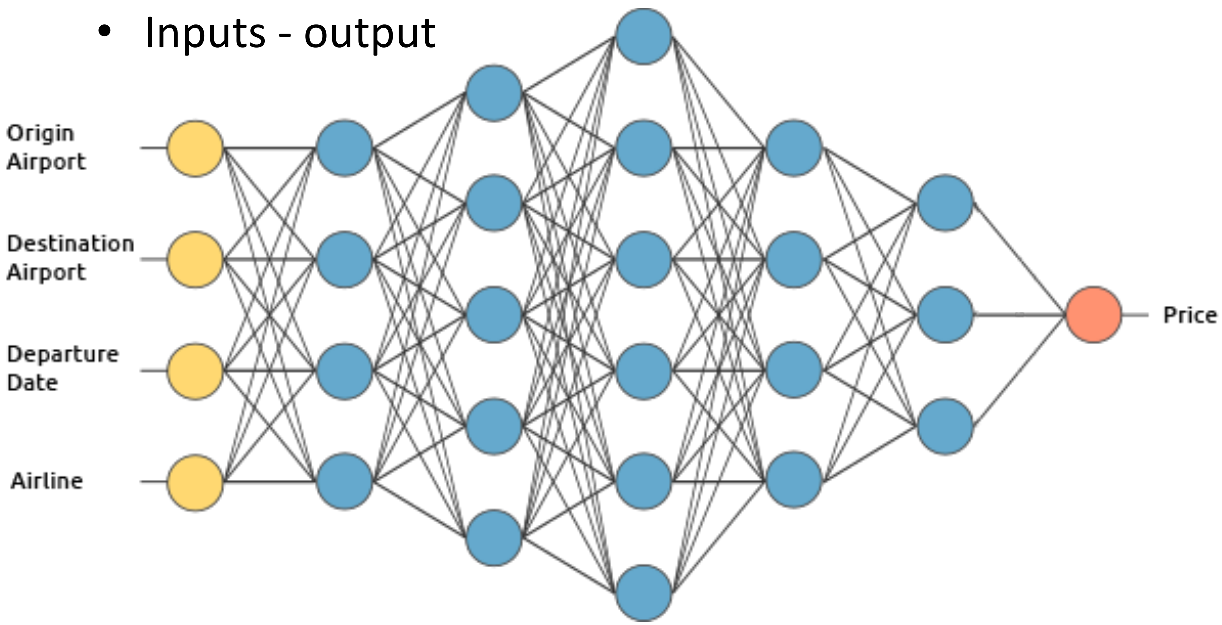


Image Segmentation Architectures

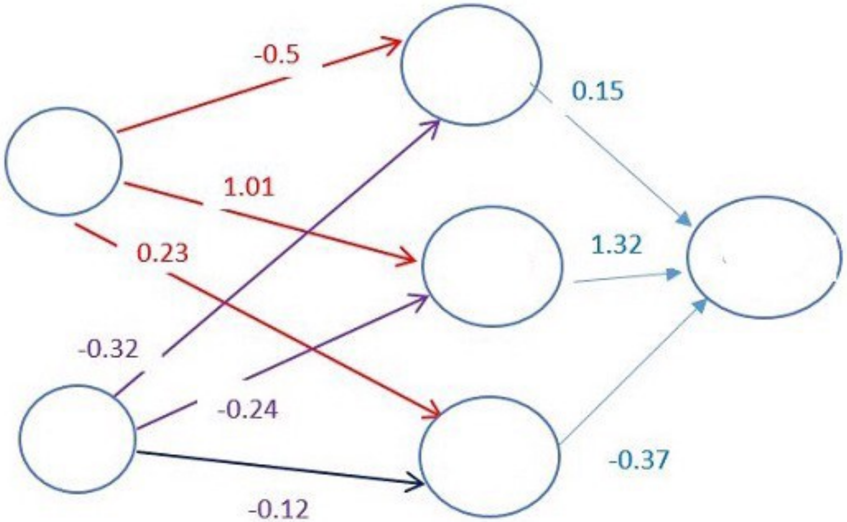
- Brief history of DL
  - Milestones
    - Artificial Neural Network
      - Inputs - output



[source: https://cs231n.github.io/neural-networks-1/]

Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Artificial Neural Network
      - Weights



[source: https://cs231n.github.io/neural-networks-1/]

Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Convolutional Neural Networks

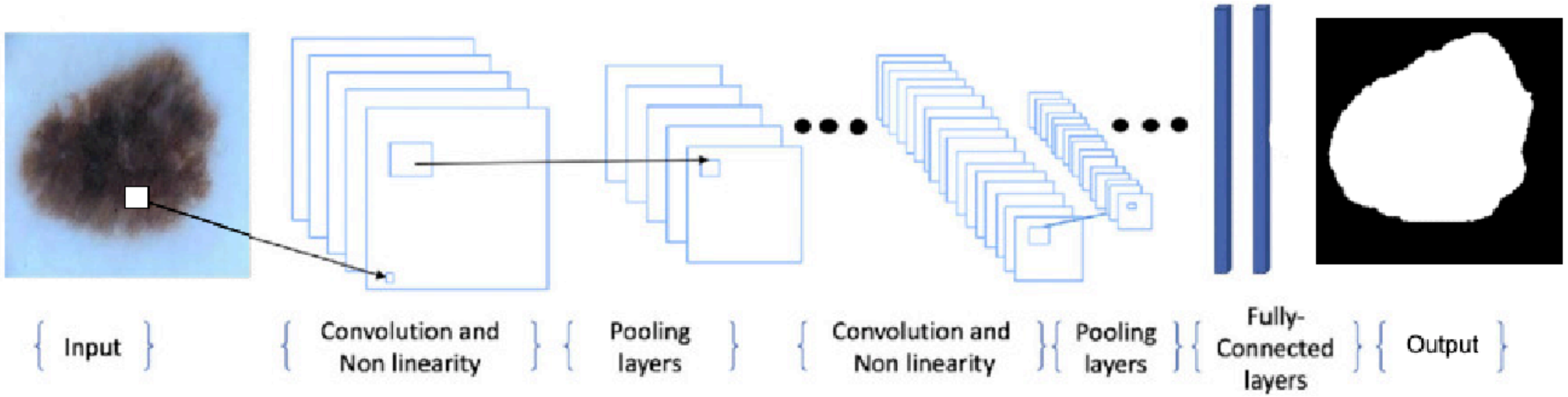
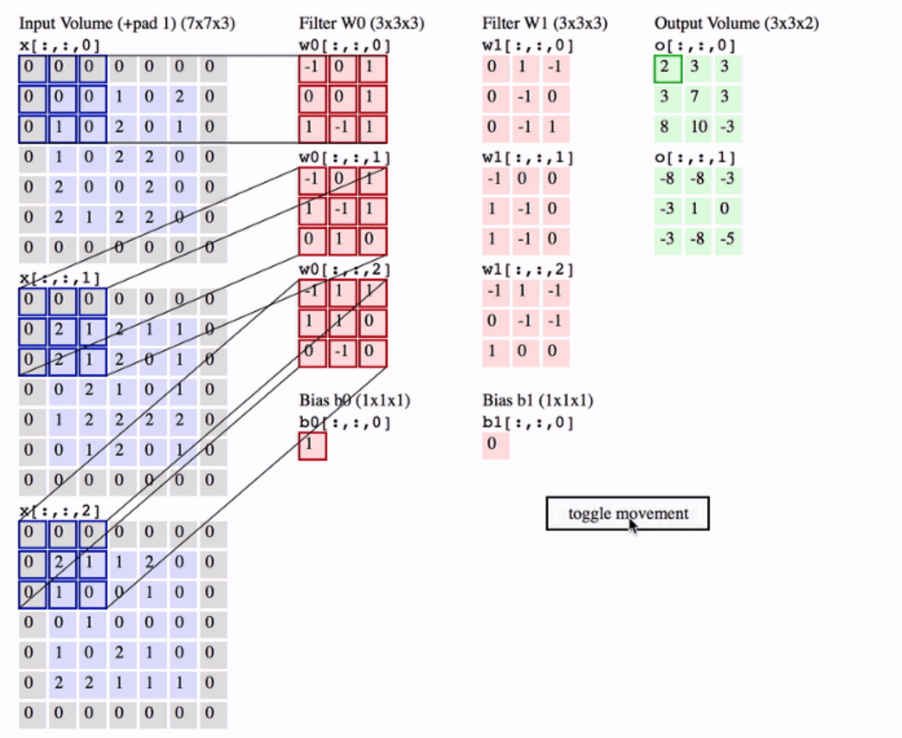


Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Convolutional Neural Networks



[source: https://cs231n.github.io/neural-networks-1/]

Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Convolutional Neural Networks

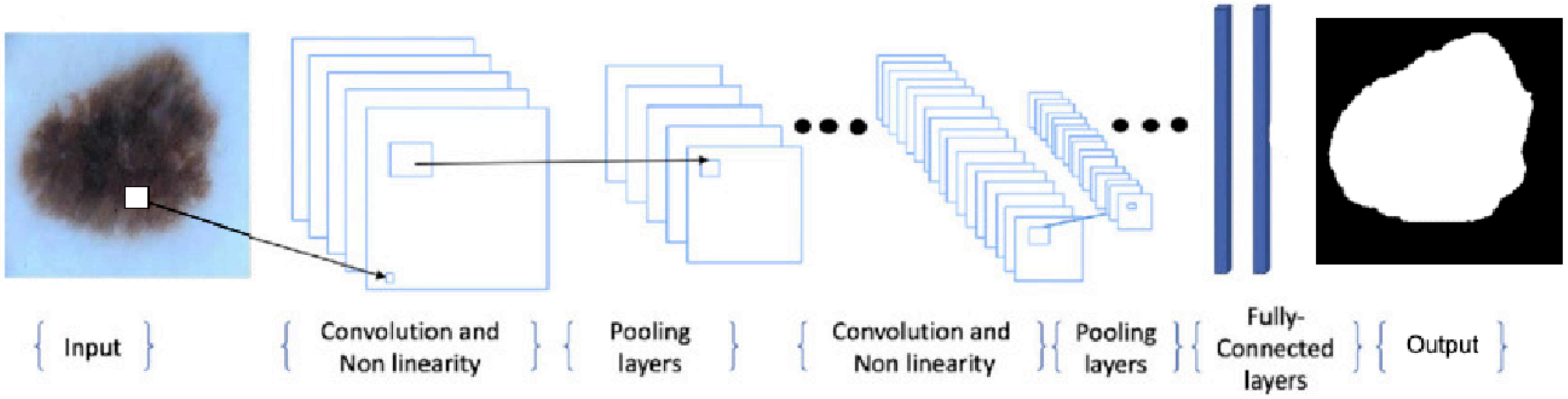
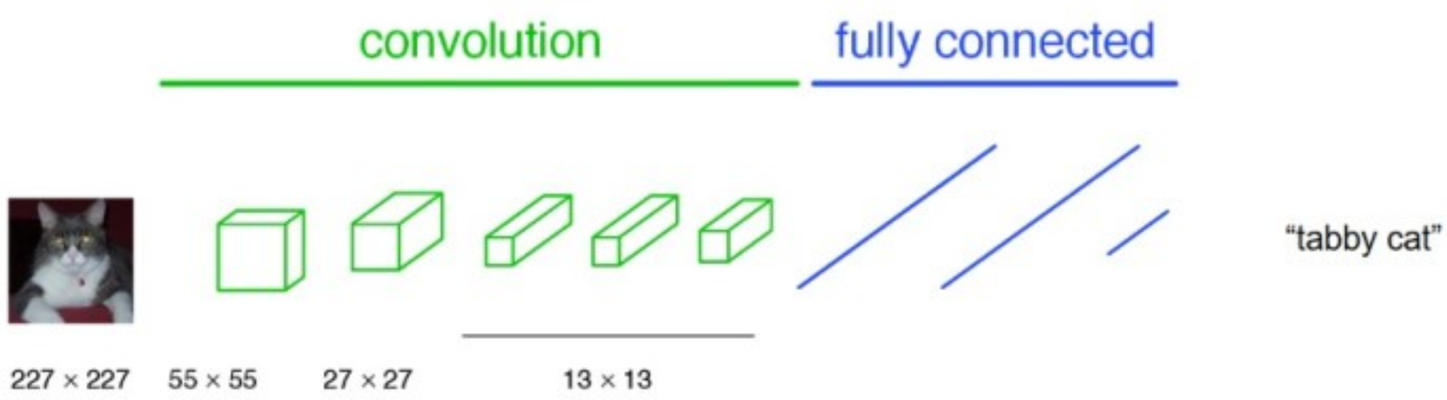


Image Segmentation Architectures

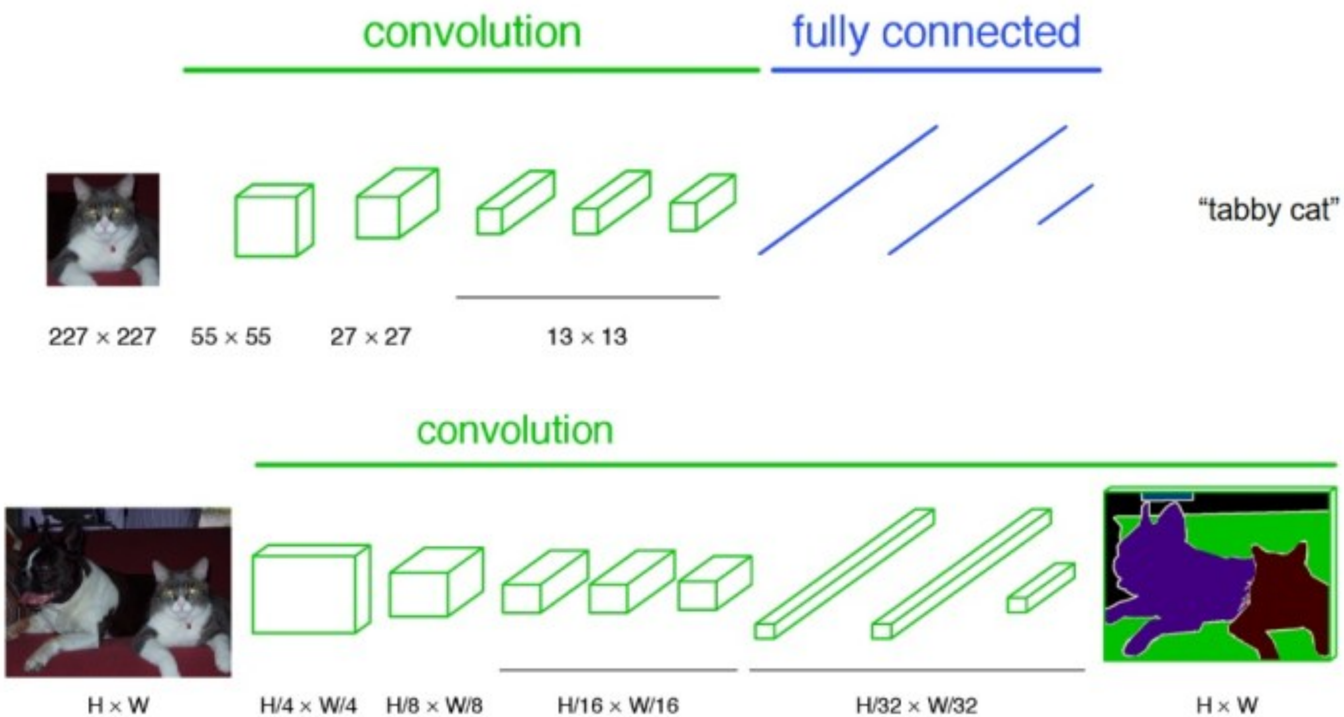
- Brief history of DL
  - Milestones
    - Fully Convolutional Networks



[source: https://yapayzeka.ai/semantik-bolutleme-fcn/]

Image Segmentation Architectures

- Brief history of DL
  - Milestones
    - Fully Convolutional Networks



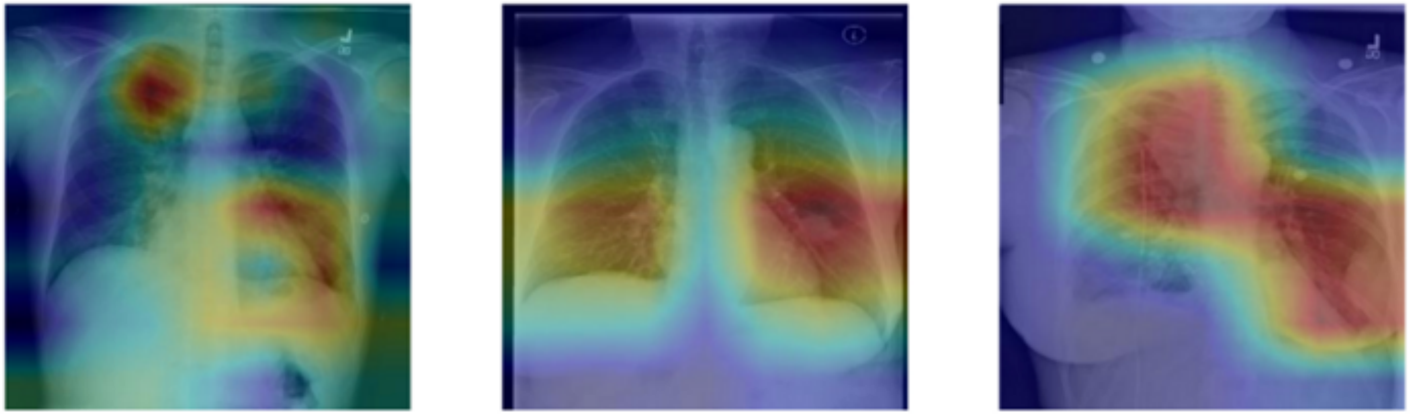
[source: https://yapayzeka.ai/semantik-bolutleme-fcn/]

Image Segmentation Architectures

- Brief history of DL
  - Is it a blackbox?
    - Not easy to follow
      - Hidden layers

Image Segmentation Architectures

- Brief history of DL
  - Is it a blackbox?
    - Not easy to follow
      - Hidden layers
    - Heat maps



[source: https://arxiv.org/abs/1610.02391]

Image Segmentation Architectures

- Image Segmentation
- Classical Methods
- Brief history of DL
- Common Image Segmentation Architectures

Image Segmentation Architectures

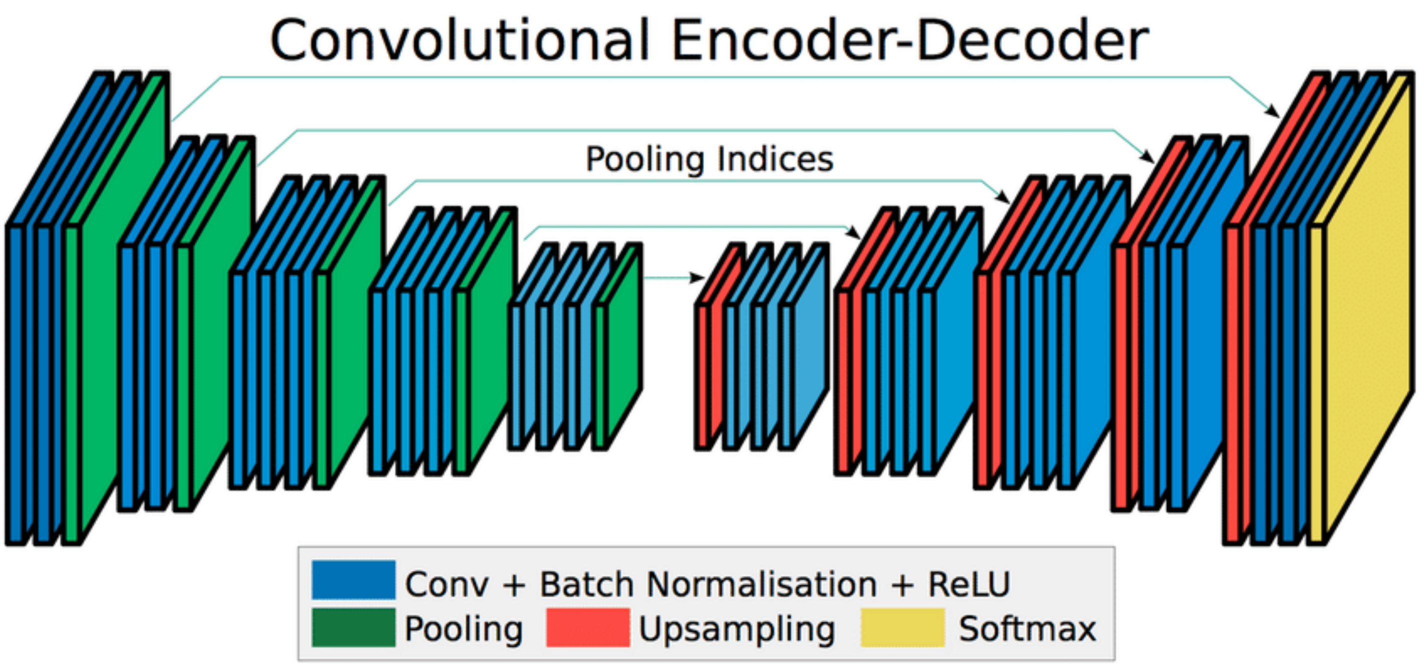
- Common Image Segmentation Architectures
  - Fully Convolutional Network
  - SegNet
  - U-Net
  - Generative Adversial Network (GAN)

Image Segmentation Architectures

- Fully Convolutional Network

Image Segmentation Architectures

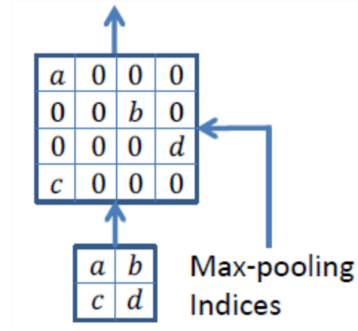
- SegNet
  - Semantic segmentation



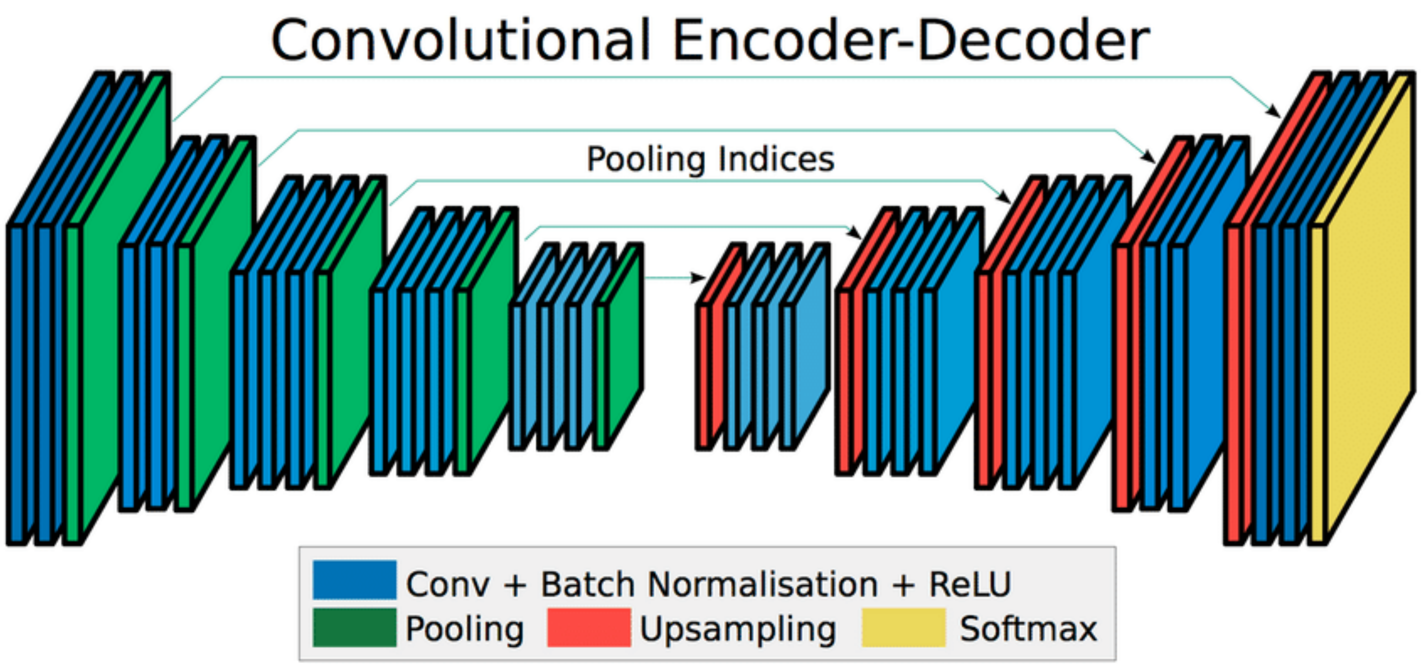
[source: Trokielewicz, Mateusz, and Adam Czajka. "Data-driven segmentation of post-mortem iris images." 2018 International Workshop on Biometrics and Forensics (IWBF). IEEE, 2018]

Image Segmentation Architectures

- SegNet
  - Semantic segmentation



Upsampling Using Max-Pooling Indices



[source: Trokielewicz, Mateusz, and Adam Czajka. "Data-driven segmentation of post-mortem iris images." 2018 International Workshop on Biometrics and Forensics (IWBF). IEEE, 2018]

Image Segmentation Architectures

- U-Net
  - Ronneberger, Fischer, and Brox, 2015
  - Medical image segmentation

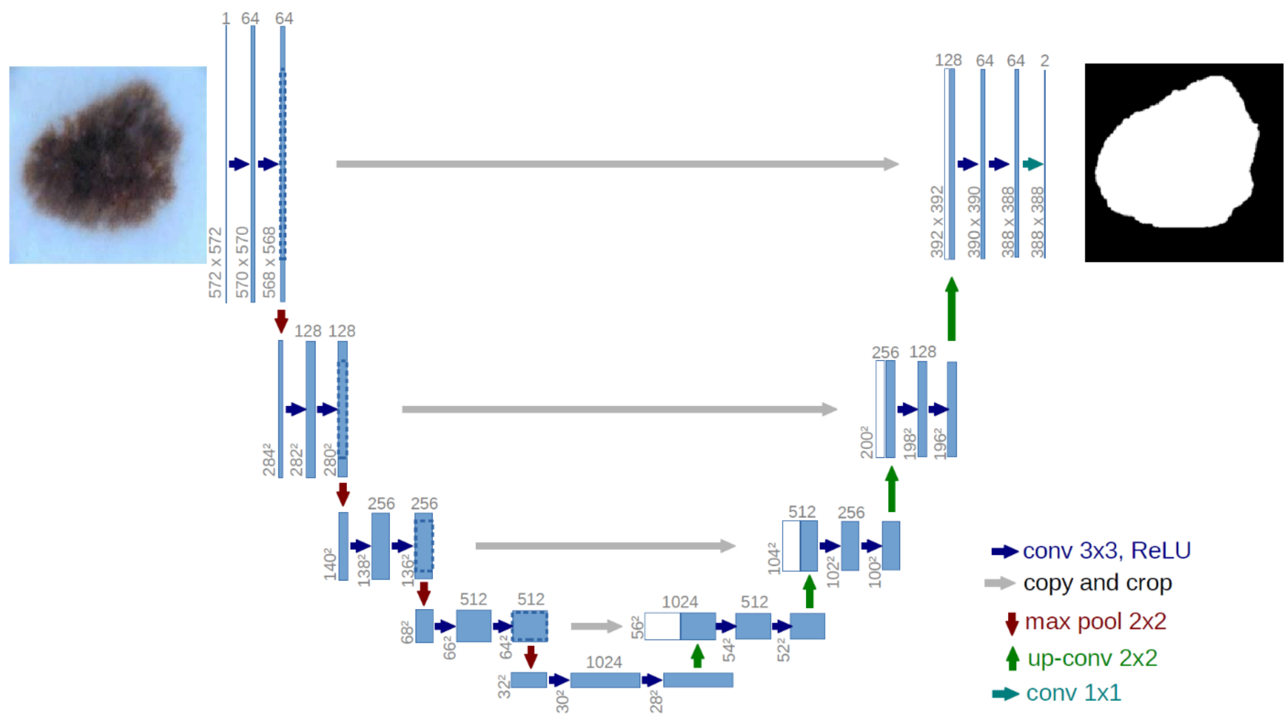


Image Segmentation Architectures

- Generative Adversial Network (GAN)
  - Goodfellow, 2014

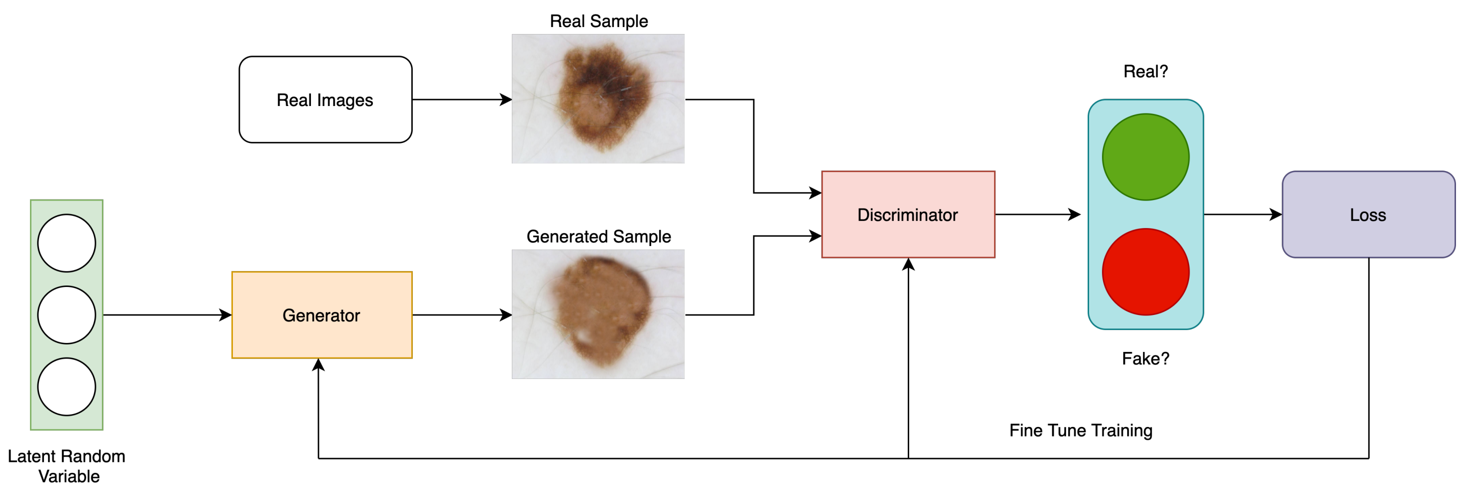
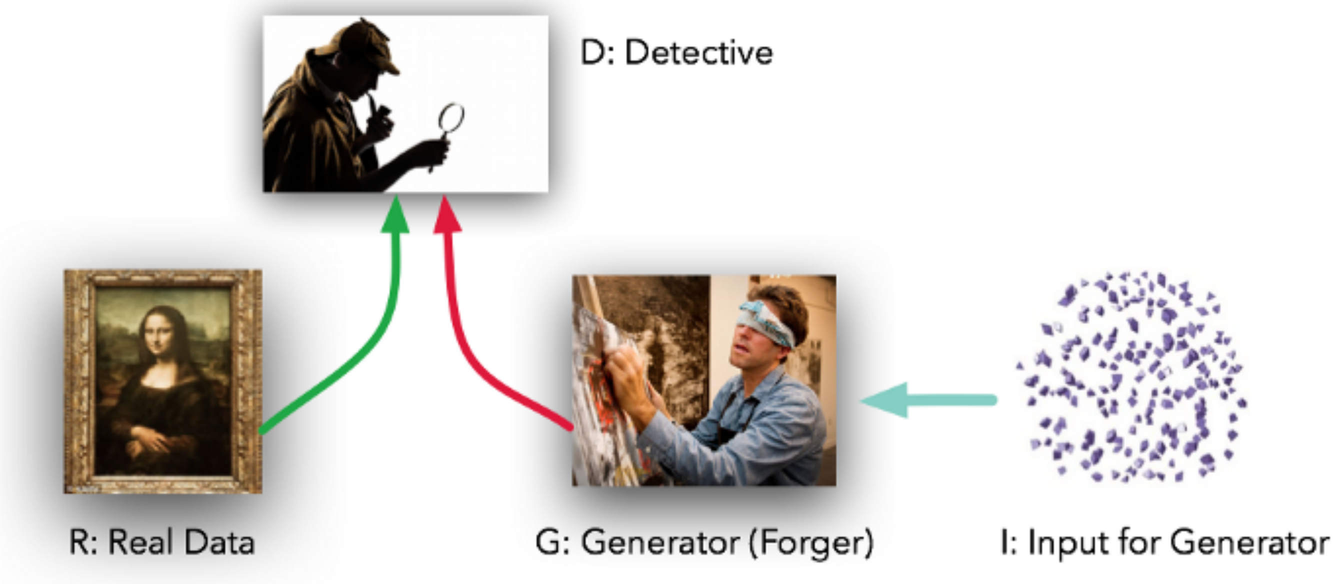


Image Segmentation Architectures

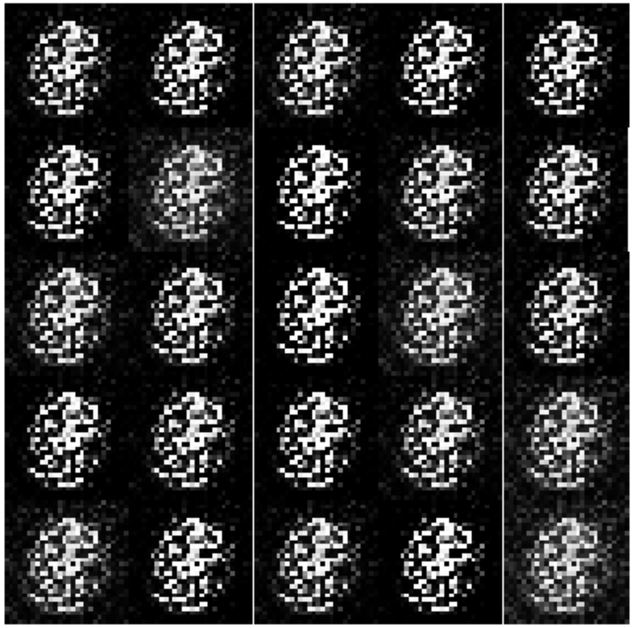
- Generative Adversial Network (GAN)
  - Goodfellow, 2014



Source : https://medium.com/@devnag/generative-adversarial-networks-gans-in-50-lines-of-code-pytorch-e81b79659e3f

Image Segmentation Architectures

- Generative Adversial Network (GAN)
  - Goodfellow, 2014



Epoch 1

Source : https://github.com/togheppi/vanilla_GAN

Outline

- Motivation
- Image Segmentation Architectures
- Methodology
- Results
- Discussions and Future Works
- Conclusion

Methodology

- Dataset
- Networks
- Tools and Frameworks
- Experiments
- Evaluation Metrics

Methodology

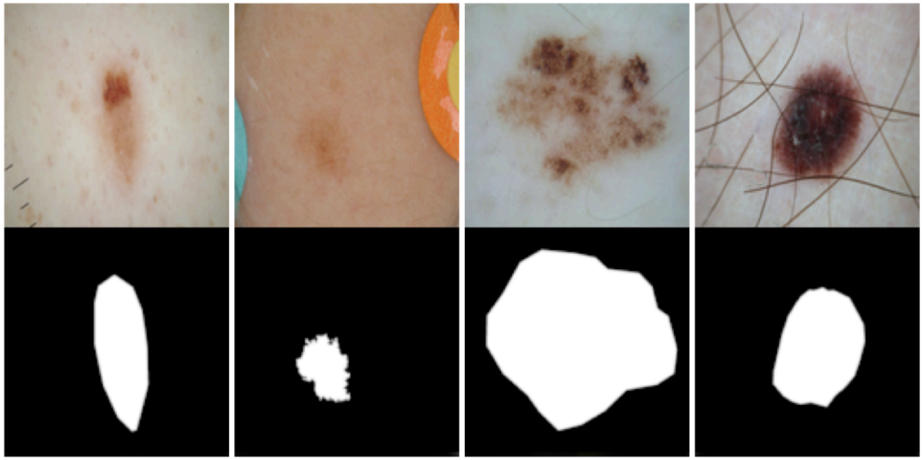
- Dataset
- Networks
- Tools and Frameworks
- Experiments
- Evaluation Metrics

Methodology

- Dataset
  - International Symposium on Biomedical Imaging (ISBI) Challenge 2017
    - Provided by The International Skin Imaging Collaboration (ISIC)

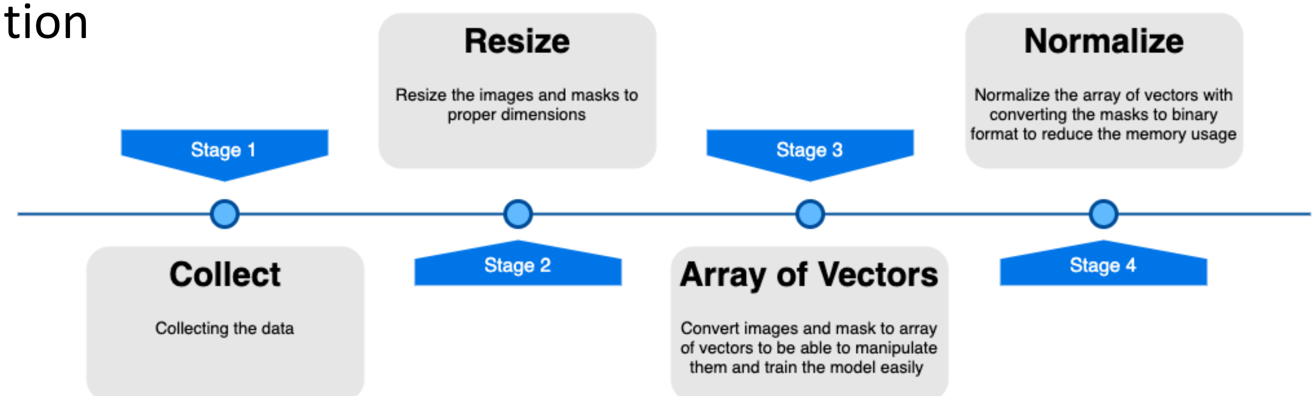
Methodology

- Dataset
  - International Symposium on Biomedical Imaging (ISBI) Challenge 2017
    - Provided by The International Skin Imaging Collaboration (ISIC)
  - 2750 **RGB images** with **binary masks** in different resolutions
    - 2000 train
    - 600 test
    - 150 validation



Methodology

- Dataset
  - International Symposium on Biomedical Imaging (ISBI) Challenge 2017
    - Provided by The International Skin Imaging Collaboration (ISIC)
  - 2750 **RGB images** with **binary masks** in different resolutions
    - 2000 train
    - 600 test
    - 150 validation



Methodology

- Dataset
- Networks
- Tools and Frameworks
- Experiments
- Evaluation Metrics

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
    - Benchmark

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?
  - MultiResUNet

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?
  - MultiResUNet
    - Not trained with the dataset that we used

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?
  - MultiResUNet
    - Not trained with the dataset that we used
    - Very new to skin lesion segmentation

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?
  - MultiResUNet
    - Not trained with the dataset that we used
    - Very new to skin lesion segmentation
  - SegAN

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?
  - MultiResUNet
    - Not trained with the dataset that we used
    - Very new to skin lesion segmentation
  - SegAN
    - GAN based segmentation architectures is not common

Methodology

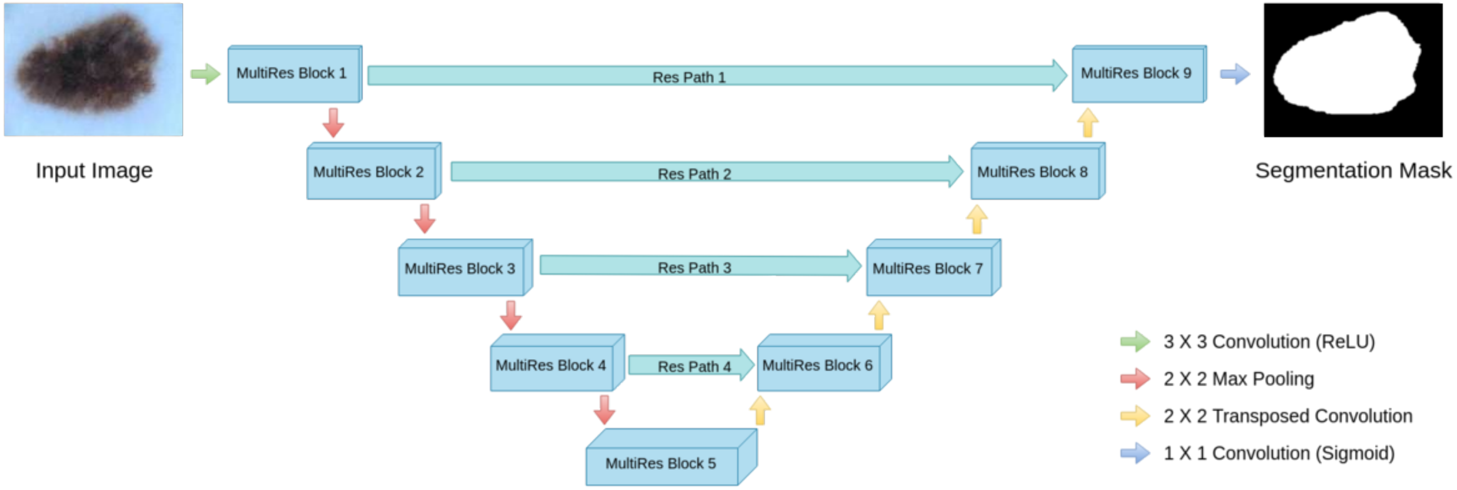
- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?
  - MultiResUNet
    - Not trained with the dataset that we used
    - Very new to skin lesion segmentation
  - SegAN
    - GAN based segmentation architectures is not common
      - The first GAN based segmentation architecture

Methodology

- Networks
  - MultiResUNet
  - SegAN
  - U-Net
- Why these networks?
  - MultiResUNet
    - Not trained with the dataset that we used
    - Very new to skin lesion segmentation
  - SegAN
    - GAN based segmentation architectures is not common
      - The first GAN based segmentation architecture
    - Not trained with the dermoscopy images

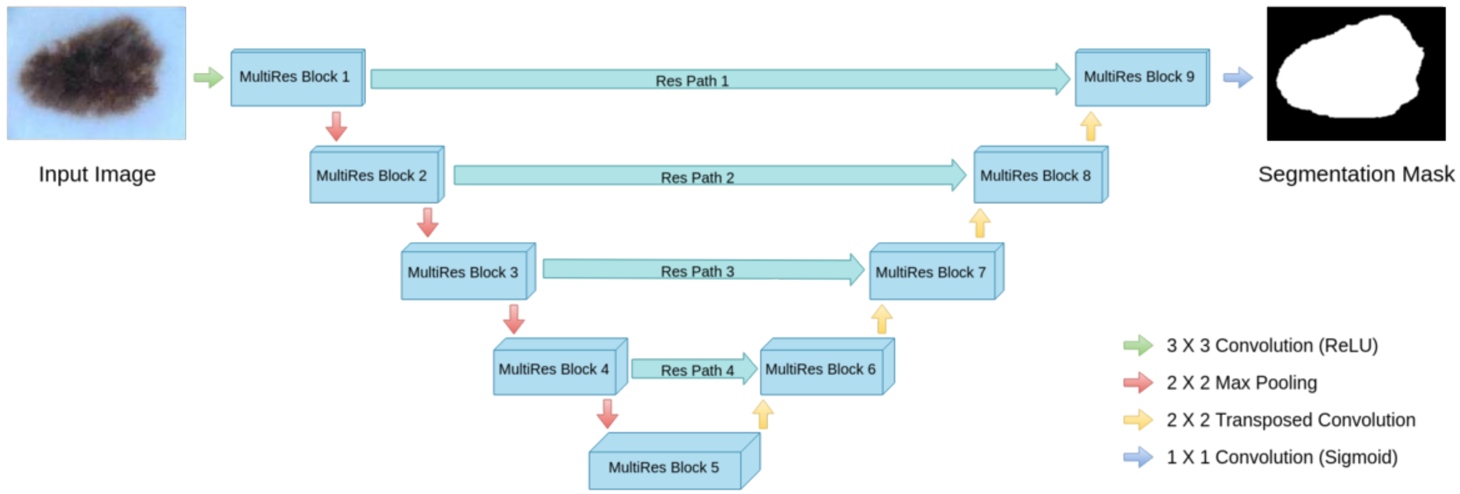
Methodology

- Networks
  - MultiResUNet



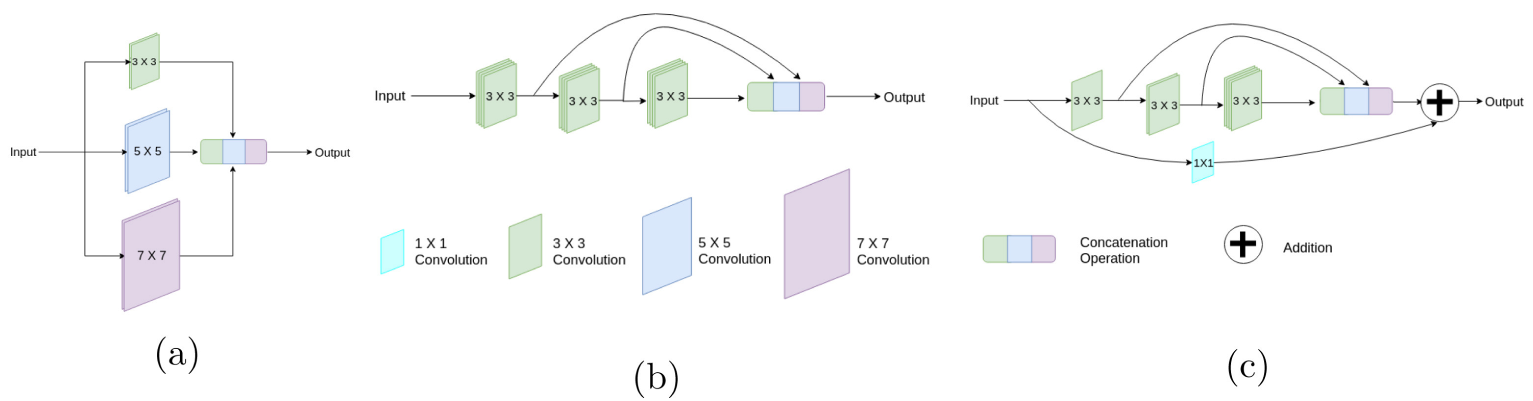
Methodology

- Networks
  - MultiResUNet
    - Updated UNet: MultiRes Blocks + Res Paths



Methodology

- Networks
  - MultiResUNet
    - MultiRes Block

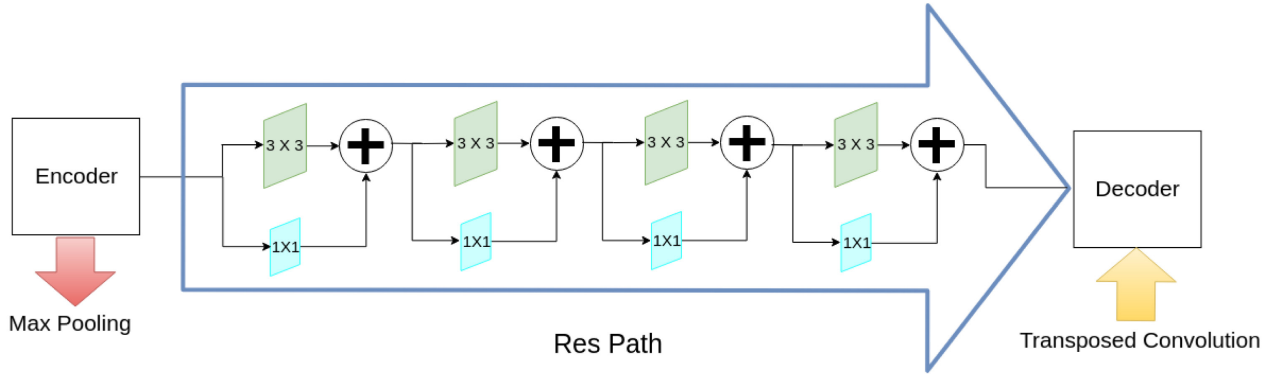


Evalution of MultiRes block : (a) Inception-like block (b) a more expensive attempt (c) MultiRes block

[source: Ibtehaz, Nabil, and M. Sohel Rahman. "MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation." Neural Networks 121 (2020): 74-87.]

Methodology

- Networks
  - MultiResUNet
    - Res Path



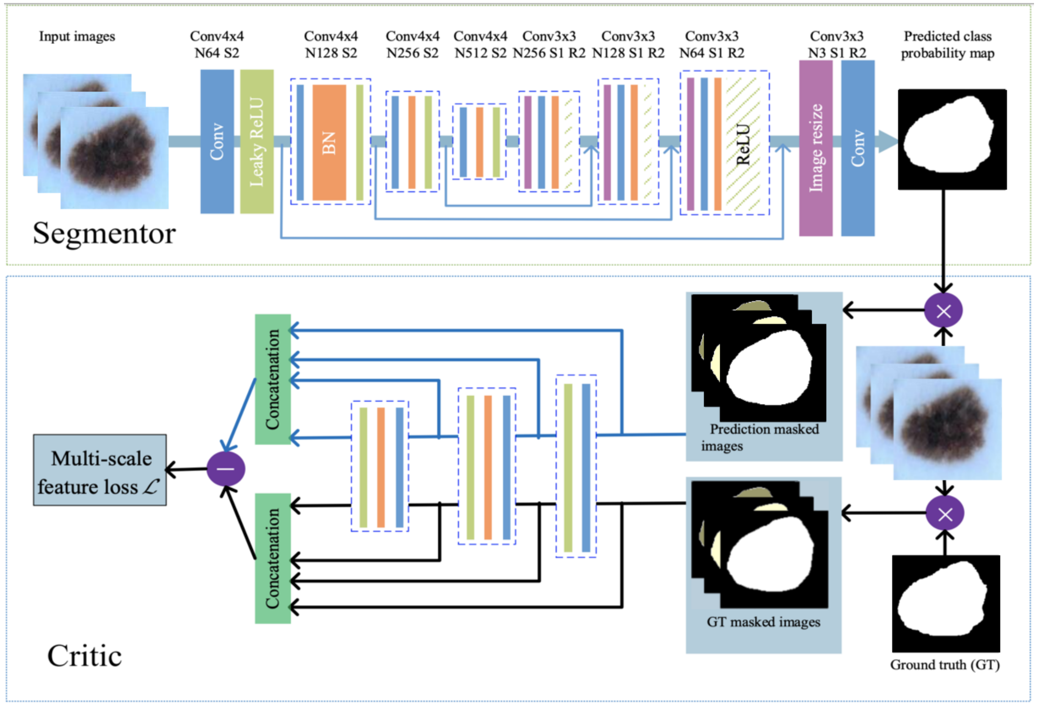
[source: Ibtehaz, Nabil, and M. Sohel Rahman. "MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation." Neural Networks 121 (2020): 74-87.]

Methodology

- Networks
  - SegAN
    - First attempt, 2016 -> no success

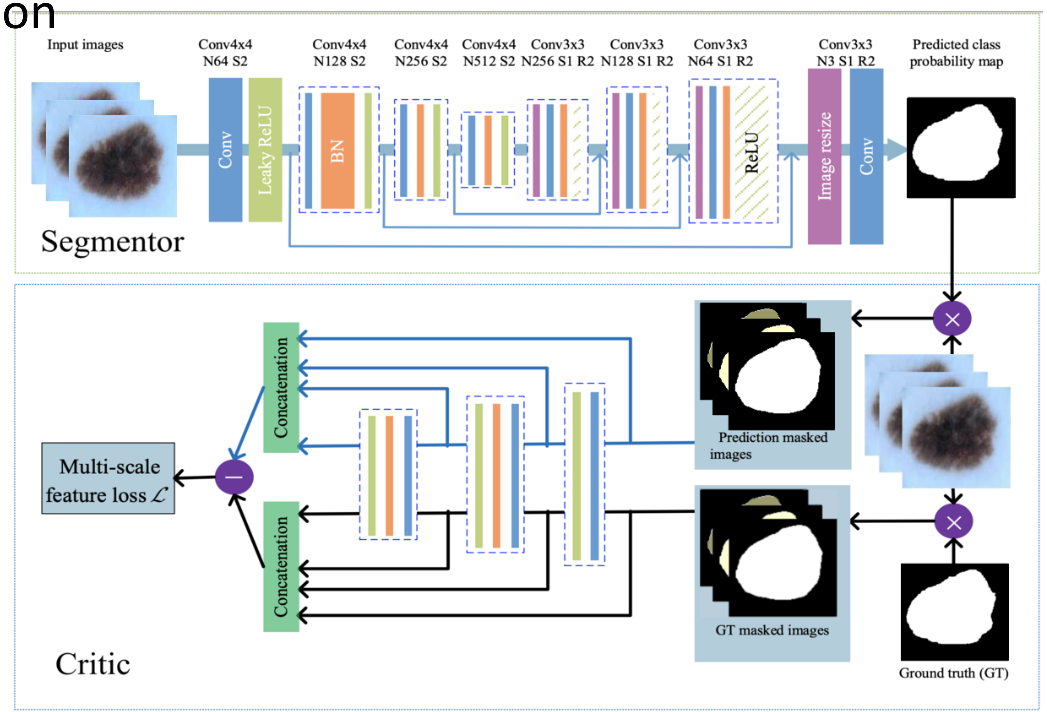
Methodology

- Networks
  - SegAN
    - First attempt, 2016 -> no success
    - Xue, Xu, Zhang, Long and Huang, 2018



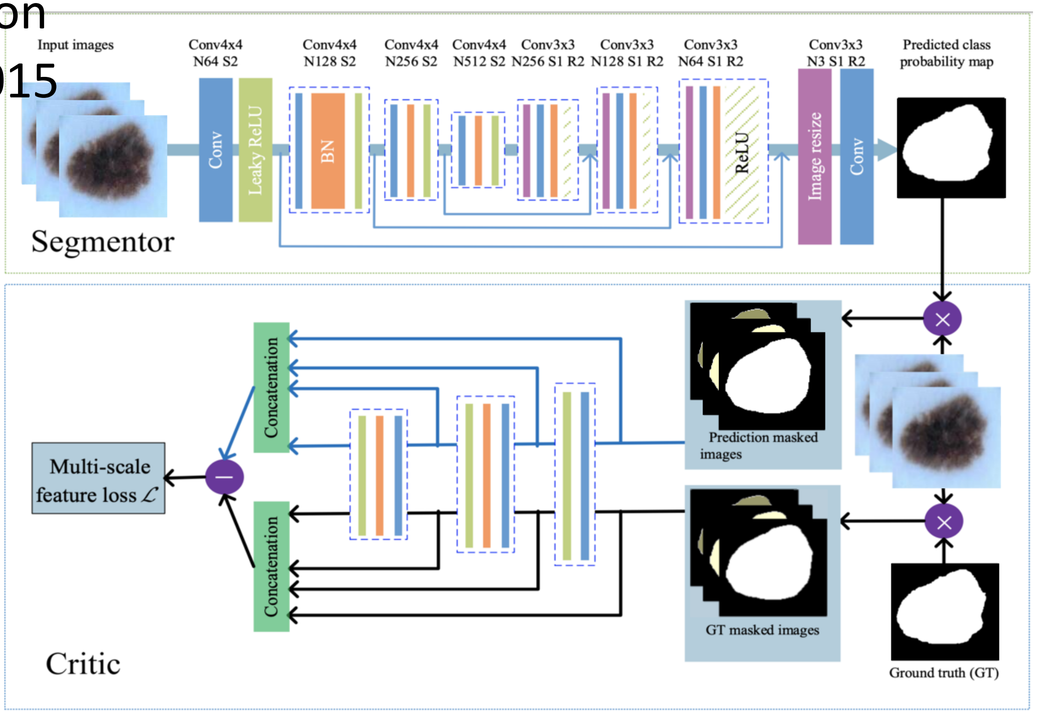
Methodology

- Networks
  - SegAN
    - First attempt, 2016 -> no success
    - Xue, Xu, Zhang, Long and Huang, 2018
    - Multi-scale loss function



Methodology

- Networks
  - SegAN
    - First attempt, 2016 -> no success
    - Xue, Xu, Zhang, Long and Huang, 2018
    - Multi-scale loss function
    - Comparet to BRATS 2015



Methodology

- Dataset
- Networks
- Tools and Frameworks
- Experiments
- Evaluation Metrics

Methodology

- Tools and Frameworks
  - Tools
    - Numpy: Scientific computing library
    - PIL: Image manipulating, filtering, masking etc.
    - ImageMagick: Image editing with rich morphological operation support
    - Jupyter Notebook: Web app that allows editing, running code with several languages

Methodology

- Tools and Frameworks
  - Deep Learning Frameworks
    - Tensorflow(v2.2.0): A library for performing numerical computations
    - Keras(v2.3.0): Neural network API for Python which can be run on top of TensorFlow or Theano
    - Pytorch(v1.5.1): Easy to use machine learning framework introduced by Facebook

Methodology

- Tools and Frameworks
  - Hardware Requirements
    - Google Colab
      - Jupyter Notebook environment
      - Runs on the cloud
      - Tesla K80 GPU
        - 25 GB of video memory
      - Ubuntu 18.04

Methodology

- Dataset
- Networks
- Tools and Frameworks
- Experiments
- Evaluation Metrics

Methodology

- A quick reminder before experiments
  - Hyperparams
    - Epoch
    - Batch size
    - Loss function
    - Optimizer
  - Early stopping
  - Gaussian noise

$$p_G(z) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z-\mu)^2}{2\sigma^2}}$$

Gaussian noise distribution in single channel image: μ and σ represents the mean value and the standard deviation, respectively

Methodology

- Experiments
  - Hyperparams
    - MultiResUNet
      - Epoch: 200
      - Batch size: 8
      - Loss function: Binary cross entropy
      - Optimizer: Adam
    - SegAN
      - Epoch: 200
      - Batch size: 8
      - Loss function: Proposed multi-scale loss function
      - Optimizer: Adam
  - Early stopping
    - After 25 epoch
  - Different Gaussian noise level
    - σ values from 0 to 50 by 10

Methodology

- Dataset
- Networks
- Tools and Frameworks
- Experiments
- Evaluation Metrics

Methodology

- Evaluation Metrics
  - Dice Coefficient (F1 score)
  - Jaccard Index
  - Accuracy
  - Sensitivity (Recall)
  - Specificity

$$Dice = \frac{2 * TP}{2 * TP + FN + FP}$$

Methodology

- Evaluation Metrics
  - Dice Coefficient (F1 score)
  - Jaccard Index
  - Accuracy
  - Sensitivity (Recall)
  - Specificity

$$Jaccard = \frac{TP}{TP + FN + FP}$$

Methodology

- Evaluation Metrics
  - Dice Coefficient (F1 score)
  - Jaccard Index
  - Accuracy
  - Sensitivity (Recall)
  - Specificity

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Methodology

- Evaluation Metrics
  - Dice Coefficient (F1 score)
  - Jaccard Index
  - Accuracy
  - Sensitivity (Recall)
  - Specificity

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

Outline

- Motivation
- Image Segmentation Architectures
- Methodology
- Results
- Discussions and Future Works
- Conclusion

Results

- Train duration
  - MultiResUnet: ~8 hours
  - SegAN: ~11 hours
  - U-Net: ~8 hours

Results

- Train duration
  - MultiResUnet: ~8 hours
  - SegAN: ~11 hours
    - Two network: Segmentor and Critic
  - U-Net: ~8 hours

Results

- Statistical metrics

Results

- Statistical metrics

MultiResUNet

Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.8169	0.7221	0.922	0.964	0.9482
10	0.7707	0.6747	0.9058	0.8923	0.9024
20	0.7395	0.624	0.8829	0.8788	0.8705
30	0.6056	0.4784	0.6056	0.8402	0.7949
40	0.4345	0.3061	0.8062	0.7033	0.8104
50	0.2851	0.1969	0.7878	0.7729	0.7847

Results

- Statistical metrics

MultiResUNet

Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.8169	0.7221	0.922	0.964	0.9482
10	0.7707	0.6747	0.9058	0.8923	0.9024
20	0.7395	0.624	0.8829	0.8788	0.8705
30	0.6056	0.4784	0.6056	0.8402	0.7949
40	0.4345	0.3061	0.8062	0.7033	0.8104
50	0.2851	0.1969	0.7878	0.7729	0.7847

SegAN

Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.8110	0.6968	0.9236	0.8998	0.9240
10	0.5570	0.4	0.8129	0.6232	0.8445
20	0.5518	0.3936	0.8134	0.6322	0.8417
30	0.5456	0.3878	0.8132	0.6329	0.8398
40	0.5378	0.3791	0.809	0.6085	0.8442
50	0.5368	0.3783	0.8115	0.6364	0.8351

Results

- Statistical metrics

MultiResUNet

Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.8169	0.7221	0.922	0.964	0.9482
10	0.7707	0.6747	0.9058	0.8923	0.9024
20	0.7395	0.624	0.8829	0.8788	0.8705
30	0.6056	0.4784	0.6056	0.8402	0.7949
40	0.4345	0.3061	0.8062	0.7033	0.8104
50	0.2851	0.1969	0.7878	0.7729	0.7847

SegAN

Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.8110	0.6968	0.9236	0.8998	0.9240
10	0.5570	0.4	0.8129	0.6232	0.8445
20	0.5518	0.3936	0.8134	0.6322	0.8417
30	0.5456	0.3878	0.8132	0.6329	0.8398
40	0.5378	0.3791	0.809	0.6085	0.8442
50	0.5368	0.3783	0.8115	0.6364	0.8351

U-Net

Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.6437	0.5343	0.8619	0.7350	0.8825
10	0.6156	0.5087	0.8559	0.7562	0.8718
20	0.5584	0.4482	0.8423	0.7680	0.8507
30	0.4746	0.3707	0.8282	0.7817	0.8316
40	0.2562	0.1737	0.7853	0.6236	0.7941
50	0.2345	0.1638	0.7954	0.7575	0.7869

Results

- Statistical metrics

MultiResUNet

Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.8169	0.7221	0.922	0.964	0.9482
10	0.7707	0.6747	0.9058	0.8923	0.9024
20	0.7395	0.624	0.8829	0.8788	0.8705
30	0.6056	0.4784	0.6056	0.8402	0.7949
40	0.4345	0.3061	0.8062	0.7033	0.8104
50	0.2851	0.1969	0.7878	0.7729	0.7847

SegAN

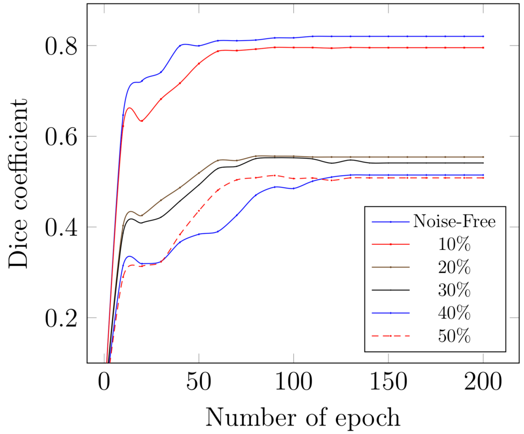
Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.8110	0.6968	0.9236	0.8998	0.9240
10	0.5570	0.4	0.8129	0.6232	0.8445
20	0.5518	0.3936	0.8134	0.6322	0.8417
30	0.5456	0.3878	0.8132	0.6329	0.8398
40	0.5378	0.3791	0.809	0.6085	0.8442
50	0.5368	0.3783	0.8115	0.6364	0.8351

U-Net

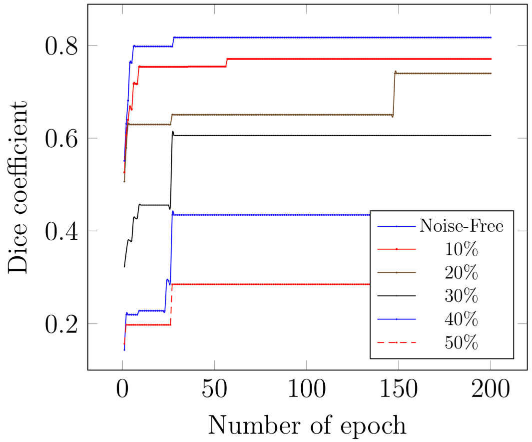
Gaussian noise(%)	Dice	Jaccard	Accuracy	Sensitivity	Specificity
0	0.6437	0.5343	0.8619	0.7350	0.8825
10	0.6156	0.5087	0.8559	0.7562	0.8718
20	0.5584	0.4482	0.8423	0.7680	0.8507
30	0.4746	0.3707	0.8282	0.7817	0.8316
40	0.2562	0.1737	0.7853	0.6236	0.7941
50	0.2345	0.1638	0.7954	0.7575	0.7869

Results

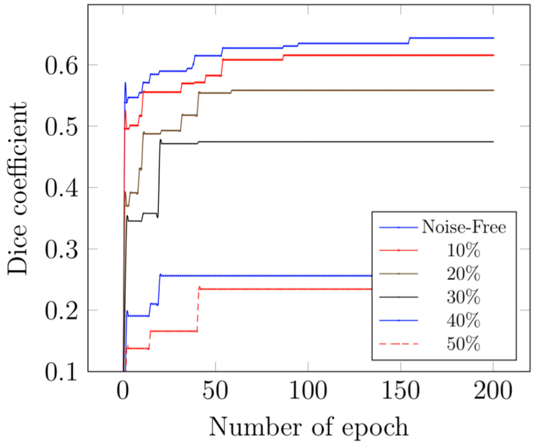
- Dice results at different Gaussian noises by number of epochs



MultiResUNet



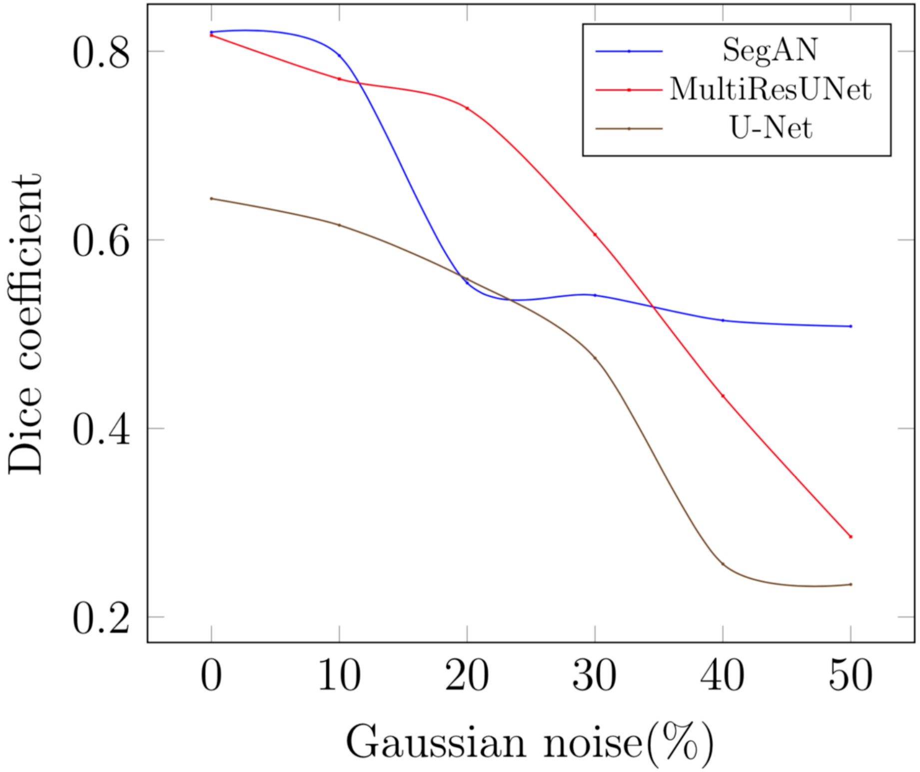
SegAN



U-Net

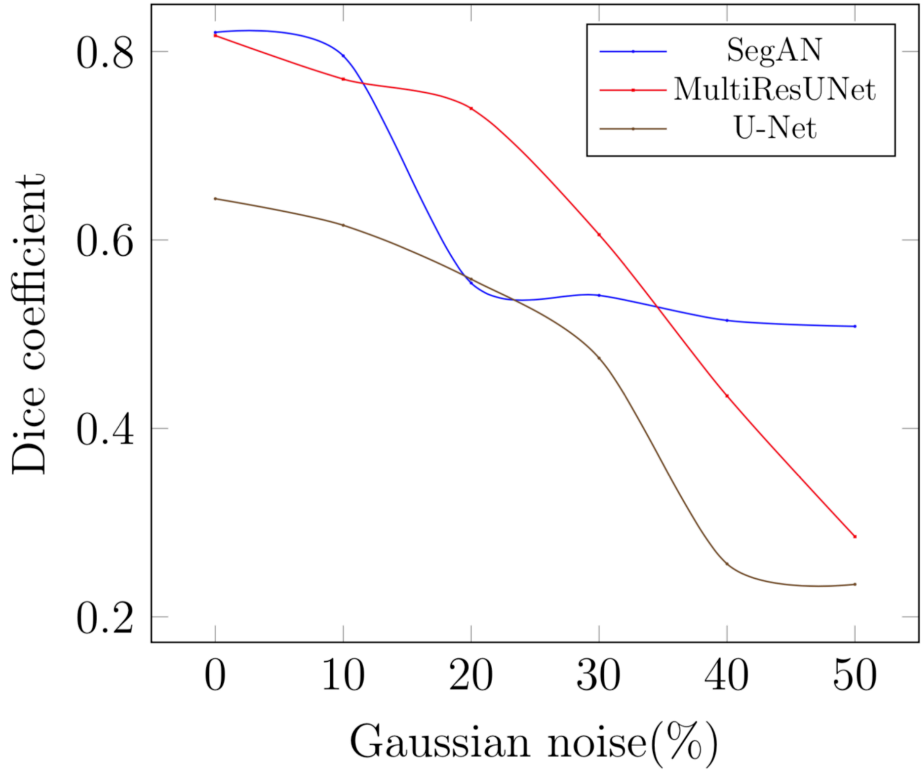
Results

- Comparison of the results of the models at different noise levels



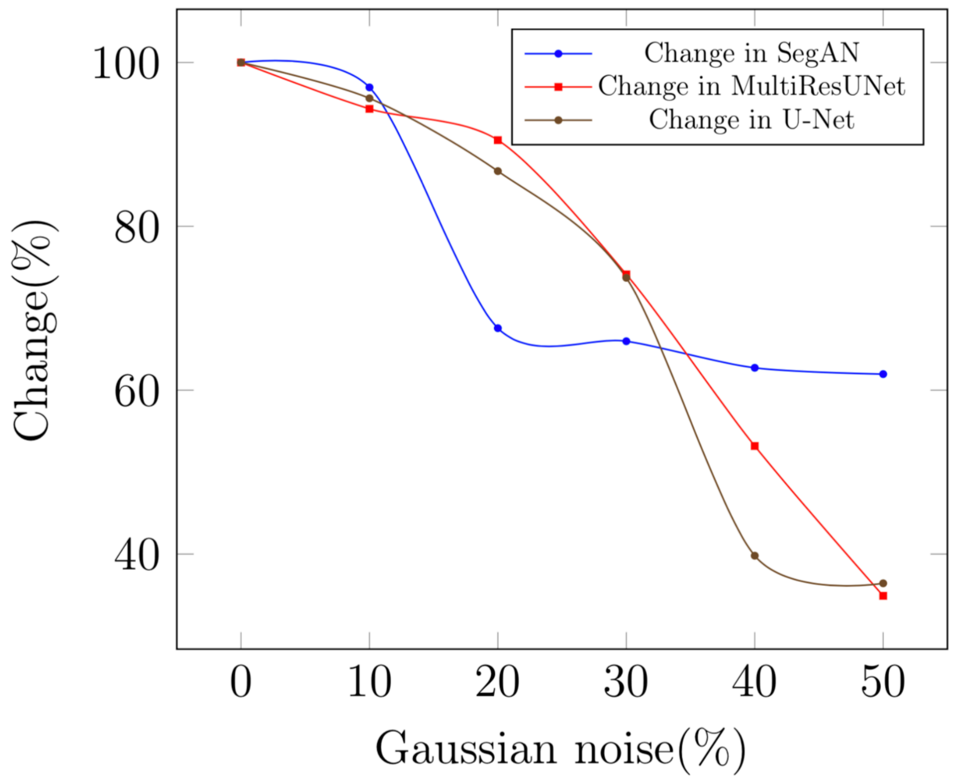
Results

- Comparison of the results of the models at different noise levels
  - The both has similar Dice for noise free images



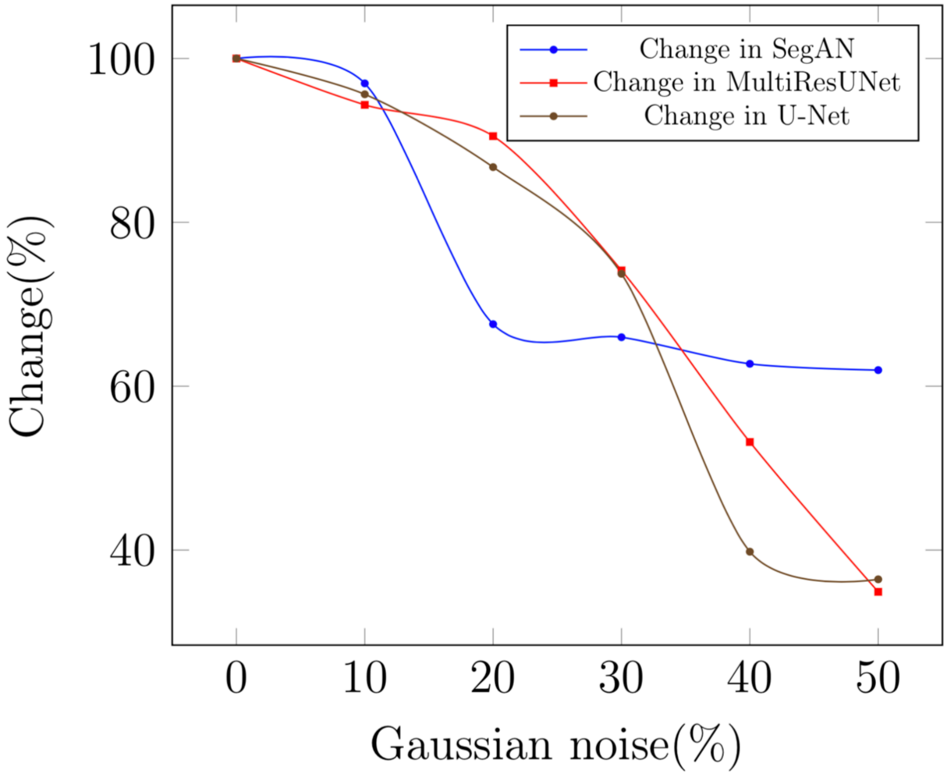
Results

- Change of success of the models by noise level



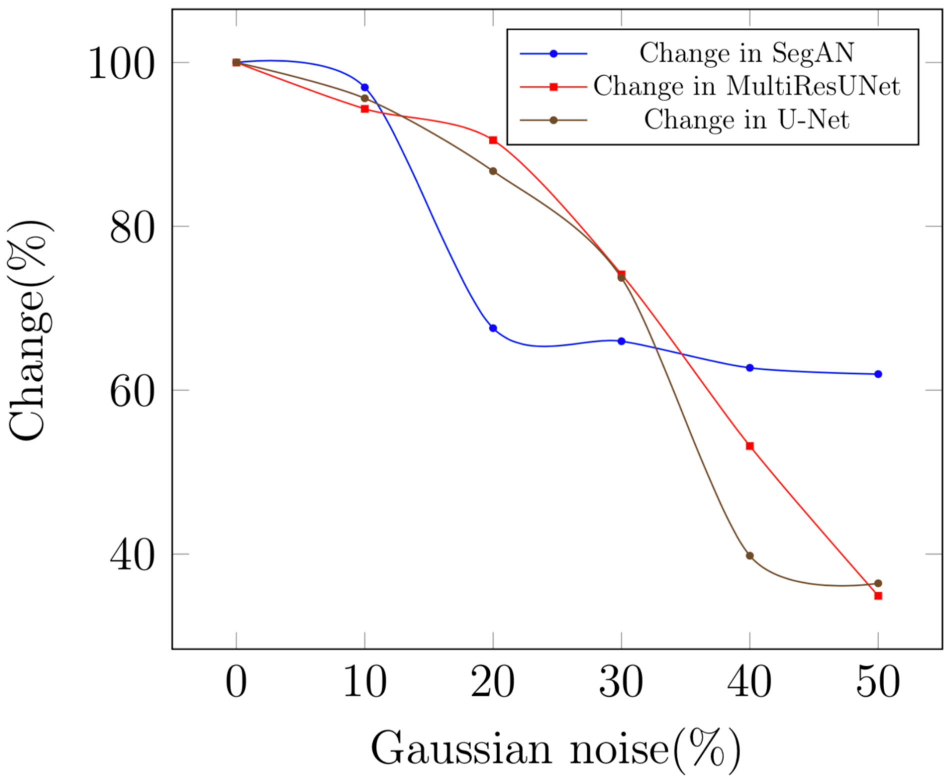
Results

- Change of success of the models by noise level
  - SegAN is more resistant against noise



Results

- Change of success of the models by noise level
  - SegAN is more resistant against noise
    - Trained with noise

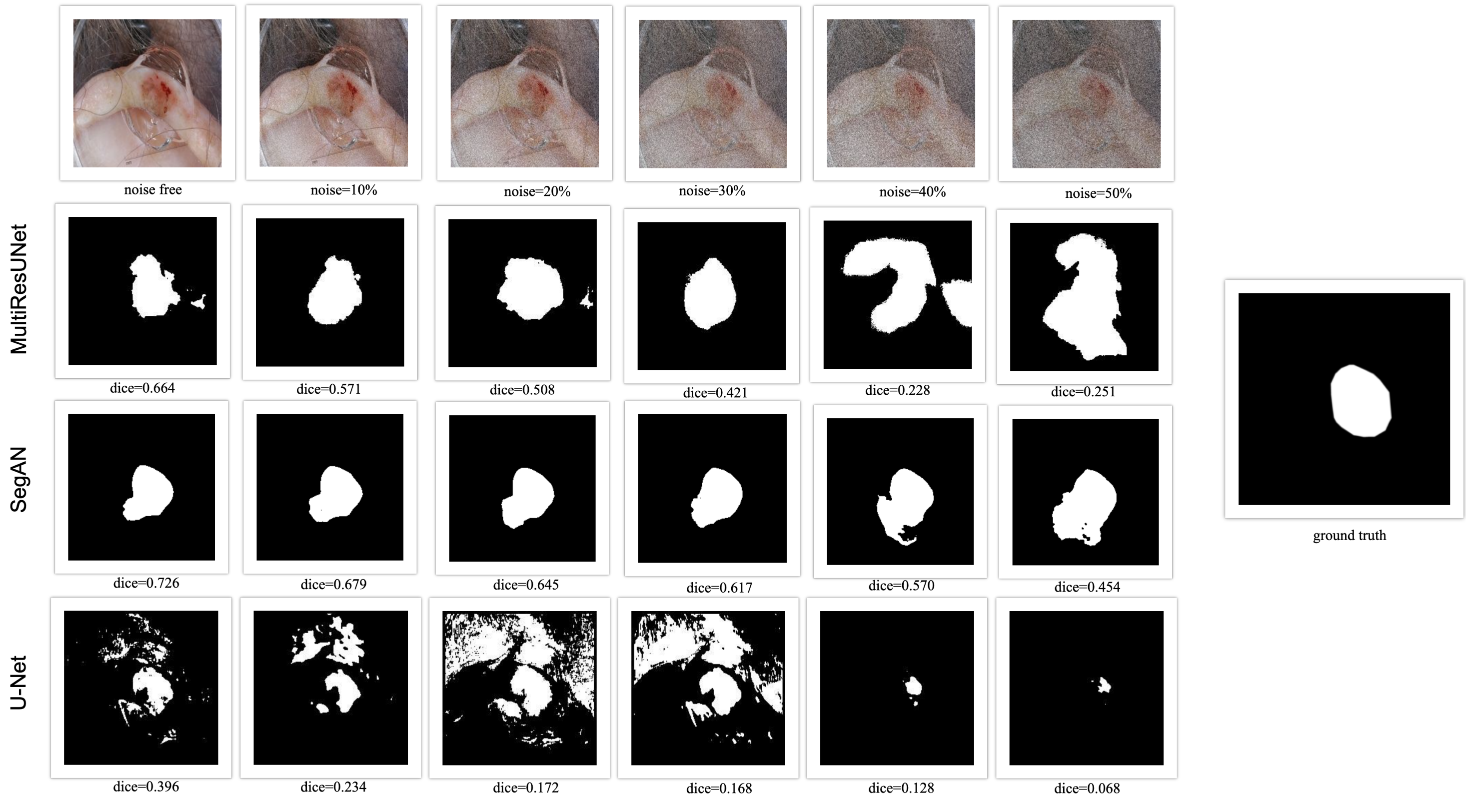


Results

- Who is the winner?

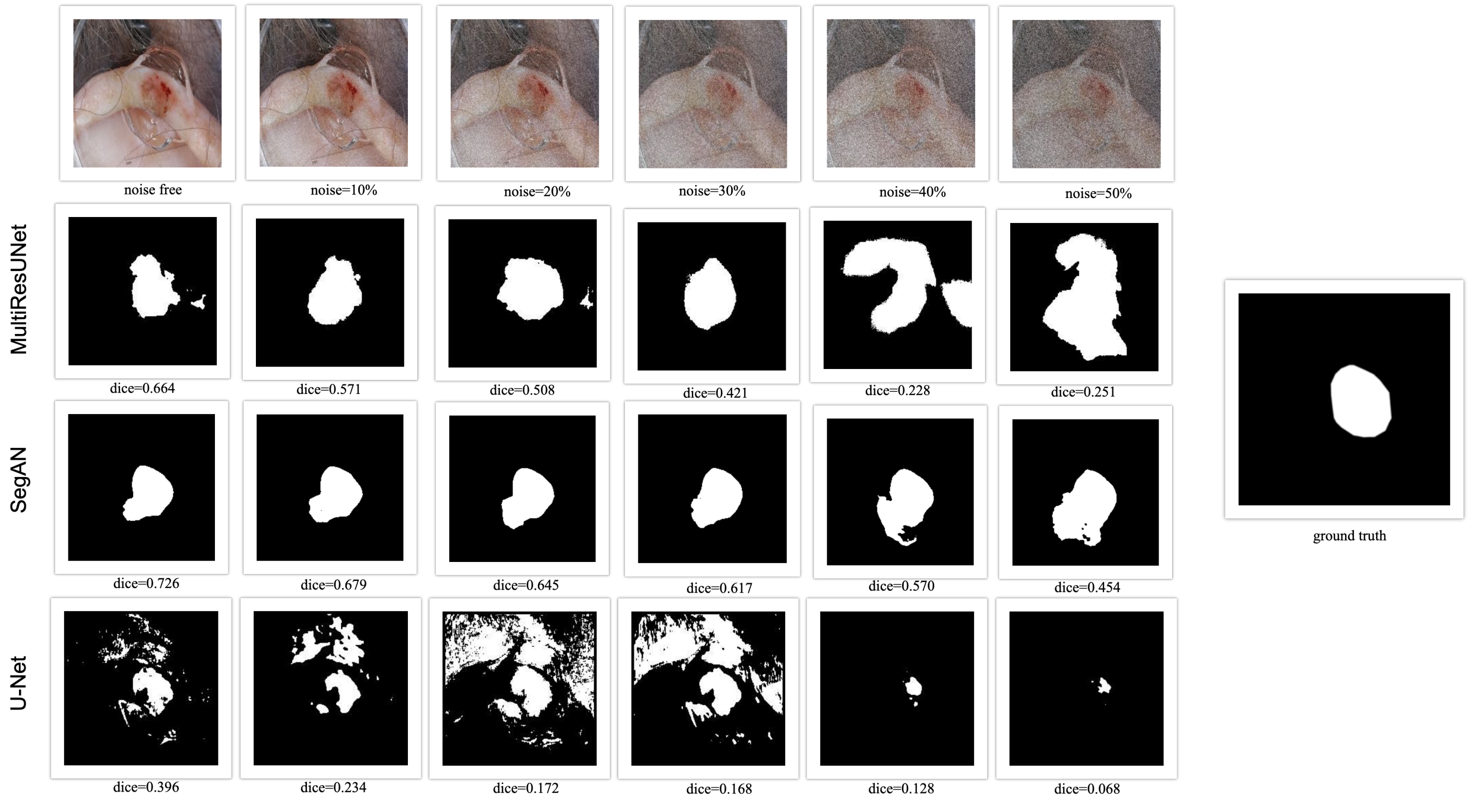
Results

- Who is the winner?



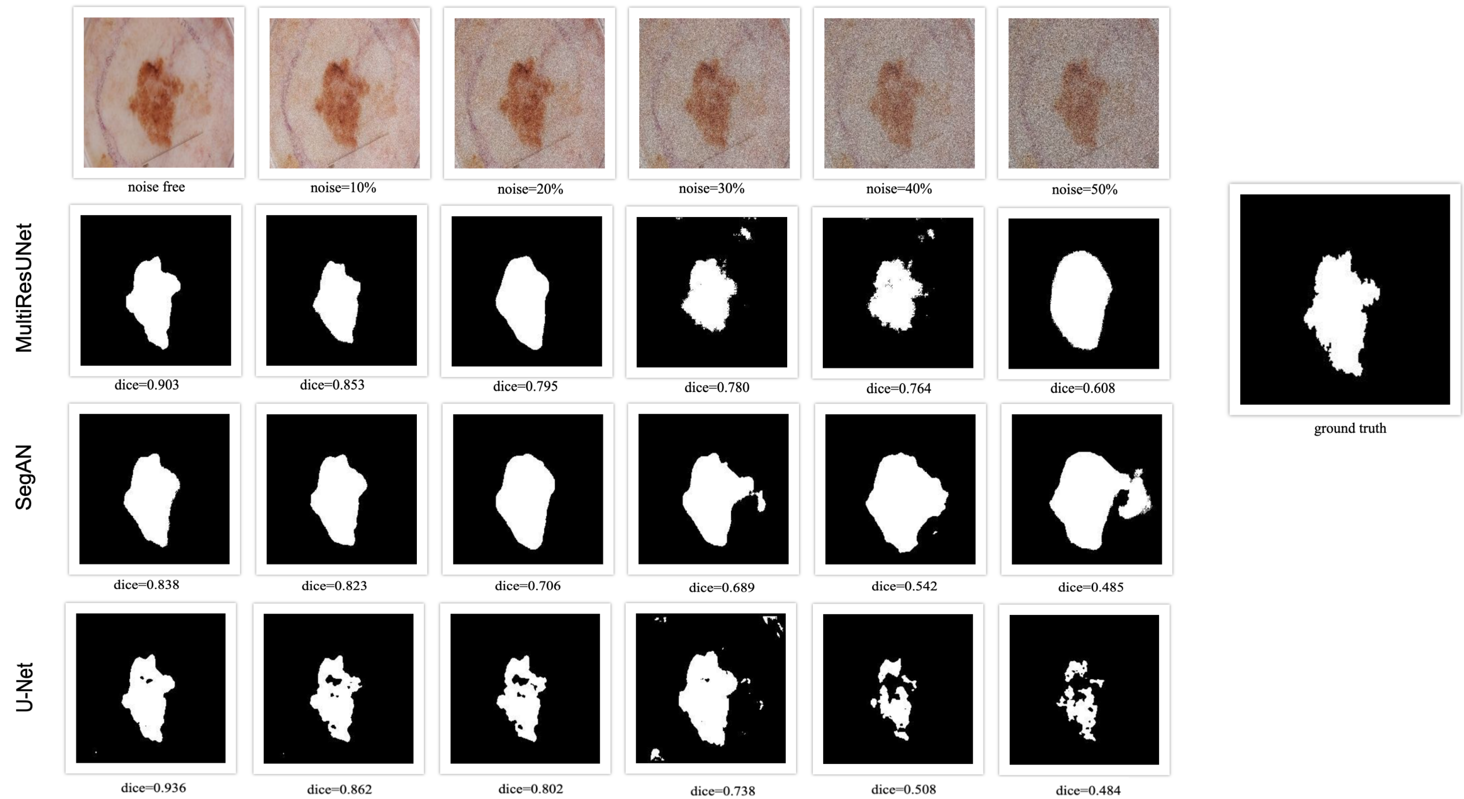
Results

- Who is the winner?



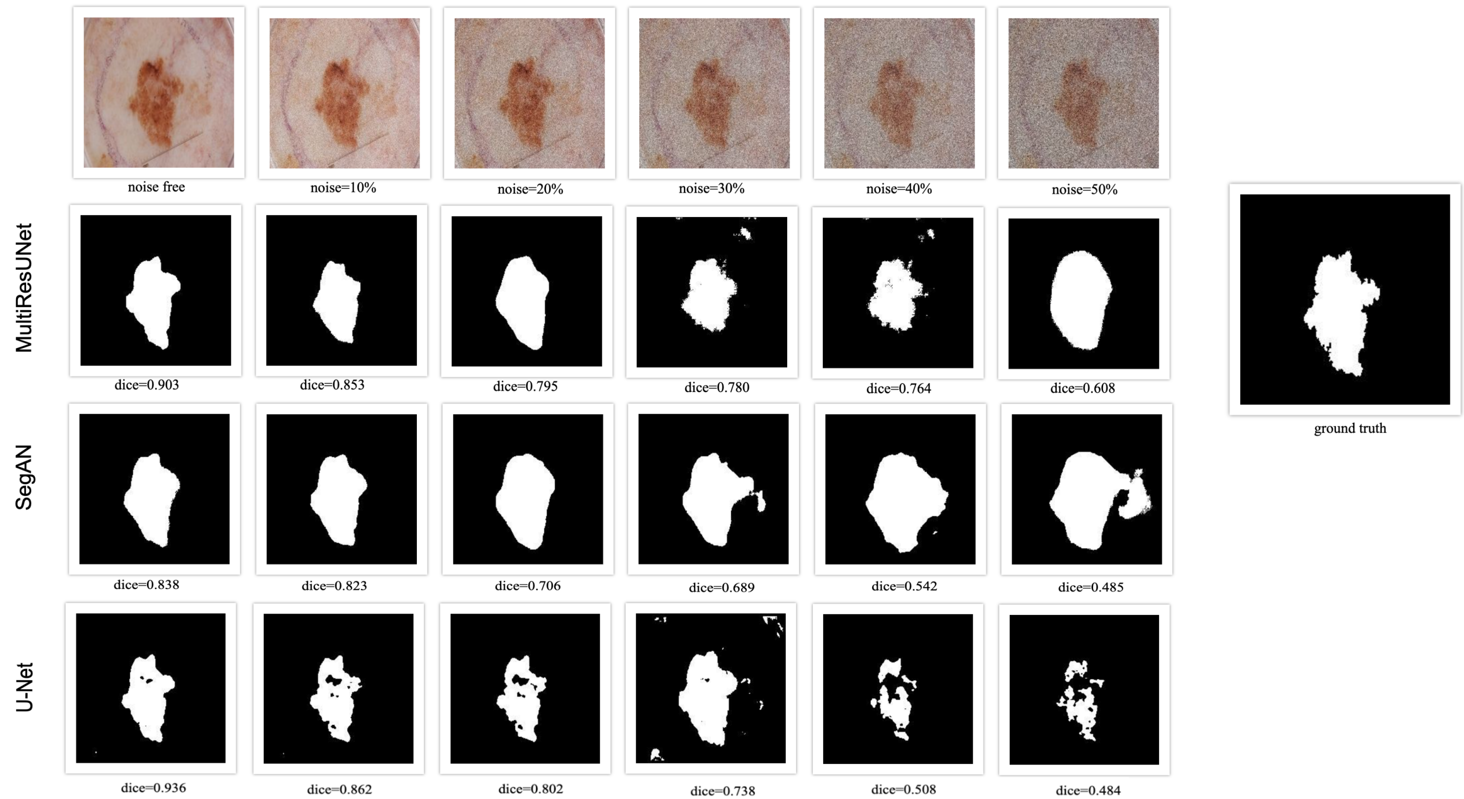
Results

- Who is the winner?



Results

- Who is the winner?



Results

- Who is the winner?
  - No silver bullet for every data

Results

- Who is the winner?
  - No silver bullet for every data
    - Trade off between models

Results

- Who is the winner?
  - No silver bullet for every data
    - Trade off between models
      - SegAN is better for noisy data
        - But takes ~11 hour to compile

Results

- Who is the winner?
  - No silver bullet for every data
    - Trade off between models
      - SegAN is better for noisy data
        - But takes ~11 hour to compile
      - MultiResUNet requires less GPU time
        - But not good at noisy images

Results

- Who is the winner?
  - No silver bullet for every data
    - Trade off between models
      - SegAN is better for noisy data
        - But takes ~11 hour to compile
      - MultiResUNet requires less GPU time
        - But not good at noisy images
      - Higher results means good model?
        - Depends on the problem definition

Results

- Who is the winner?
  - No silver bullet for every data
    - Trade off between models
      - SegAN is better for noisy data
        - But takes ~11 hour to compile
      - MultiResUNet requires less GPU time
        - But not good at noisy images
      - Higher results means good model?
        - Depends on the problem definition
  - No precise superiority of the models to each other for certain data

Outline

- Motivation
- Image Segmentation Architectures
- Methodology
- Results
- Discussions and Future Works
- Conclusion

Discussions & Future Works

- Missing large dataset

Discussions & Future Works

- Missing large dataset
  - More data augmentation
    - GAN: fake image generation architecture

Discussions & Future Works

- Missing large dataset
  - More data augmentation
    - GAN: fake image generation architecture
- Add more noise to relevant points of the models
  - E.g. Activation functions, loss functions, weights, or hidden layers

Discussions & Future Works

- Missing large dataset
  - More data augmentation
    - GAN: fake image generation architecture
- Add more noise to relevant points of the models
  - E.g. Activation functions, loss functions, weights, or hidden layers
- Combine the models
  - MultiResUNet as the segmentor network of SegAN

Discussions & Future Works

- Missing large dataset
  - More data augmentation
    - GAN: fake image generation architecture
- Add more noise to relevant points of the models
  - E.g. Activation functions, loss functions, weights, or hidden layers
- Combine the models
  - MultiResUNet as the segmentor network of SegAN
- Visualize with the help of heat maps

Outline

- Motivation
- Image Segmentation Architectures
- Methodology
- Results
- Discussions and Future Works
- Conclusion

Conclusion

- This thesis addressed to
  - Evaluate RGB dermoscopy images in various dimensions
  - Adapt state of the art models to skin lesion problem
    - MultiResUNet, SegAN
  - Test the accuracy in non trained noisy data
- This thesis showed that
  - Preferred models get remarkable results on the selected dataset
    - ISBI 2017 Challenge
  - SegAN is more resistant to noise against MultiResUNet
  - MultiResUNet can be preferred for noise free data