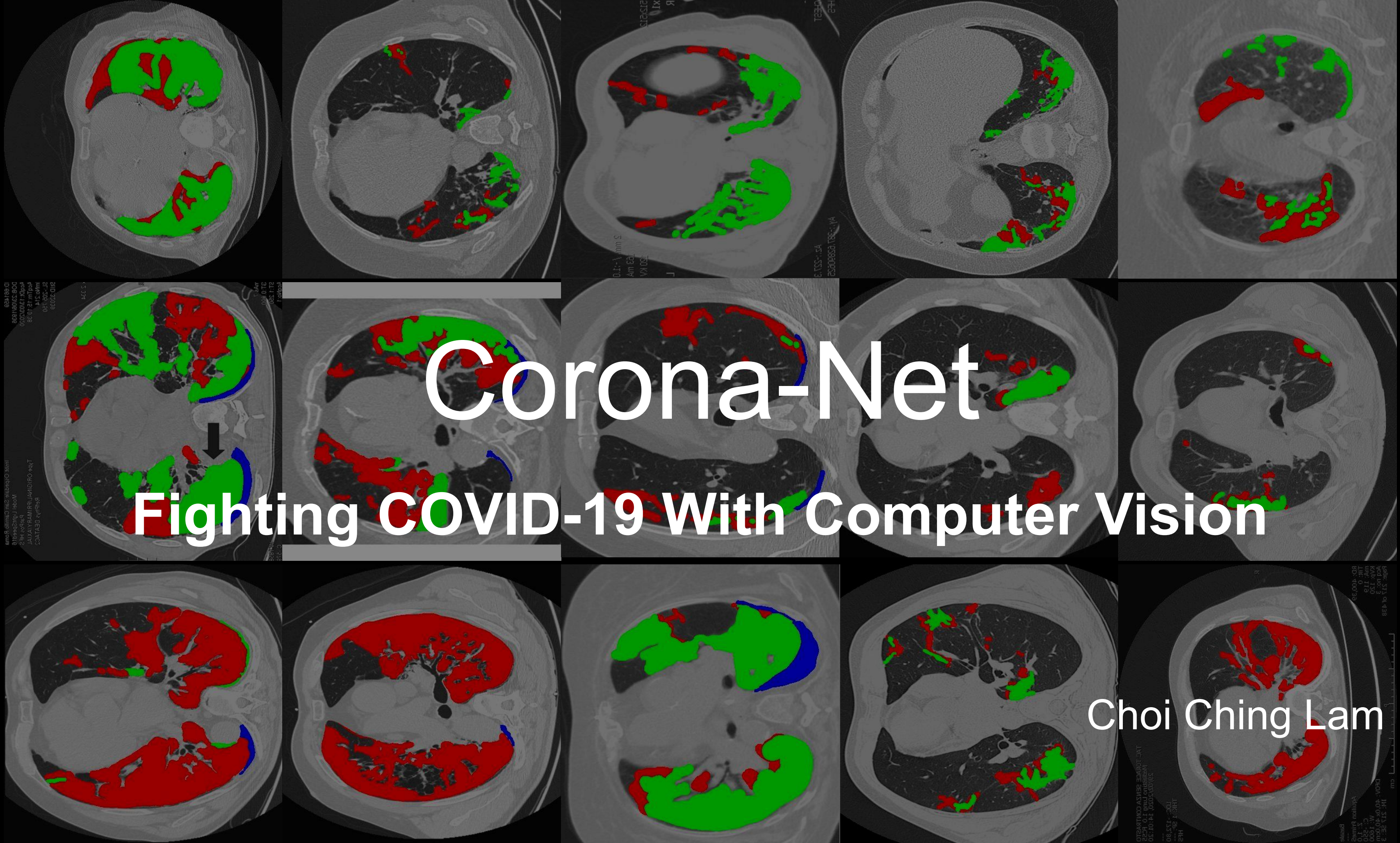


Corona-Net

Fighting COVID-19 With Computer Vision

Choi Ching Lam

Self Intro

- Choi Ching Lam
- 17 year old, Form 5 student from Hong Kong
- Favourite languages: Python, Julia
- Currently interning at NVIDIA’s AI Tech Center
- Into Computer Vision, aspires to become a researcher
- https://github.com/chinglamchoi
- https://www.linkedin.com/in/ching-lam-choi-7609541a0/
- https://medium.com/@cchoi314

Background

- Inspired by doctors from Wuhan on TV
- Inspired by Johns Hopkins University's (Center for Systems Science and Engineering (CSSE)) COVID-19 Dashboard
- Relevant to previous work on brain tumour boundary resection for lower grade glioma

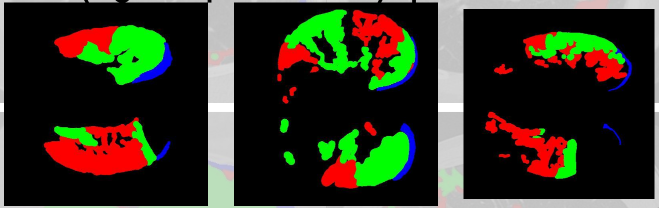
Problem Statement

- Hospitals are overwhelmed with COVID-19 patients
  - Manpower shortage → Doctors (esp radiologists), etc
  - Supplies shortage → ventilators, masks, etc
- **Solution**: **Automate CT diagnosis confirmation with AI**
  - Determine severity → Triage patients, allocate supplies
  - Gauge mortality probability
  - (Future) Design personalised treatment

Corona-Net

1. **Binary Classification**
    a. Infected (1) / not-infected (0) with COVID-19
2. **Binary Segmentation:**

    a. Predict all infected (symptoms) pixels of COVID-19 in CT
3. **3-Class Segmentation:**



    a. Predict all infected pixels & type (1 in 3) of symptoms:
    ground glass, consolidation, pleural effusion

Technologies Used

- Language: Python with NumPy library
- AI library: PyTorch
- Image processing libraries: Albumentations, Torchvision, Scikit-image, Matplotlib

Python with NumPy

- NumPy: parallelism & vectorisation
- NumPy: better support for matrices & tensors & operations
- Easy to prototype with, elegant syntax
- Powerful libraries

What to use for Image Processing?

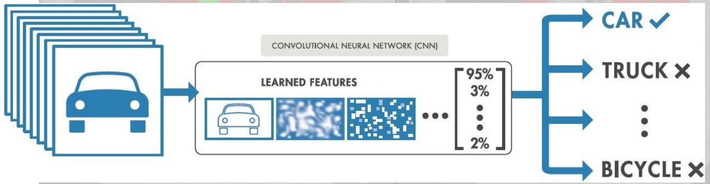
- Matplotlib vs. Scikit-image vs. Torchvision vs. Albumentations
- Matplotlib: General purpose
- Scikit-image: Advanced algorithms
- Torchvision: Tight integration with PyTorch
- Albumentations: Biomedical Imaging

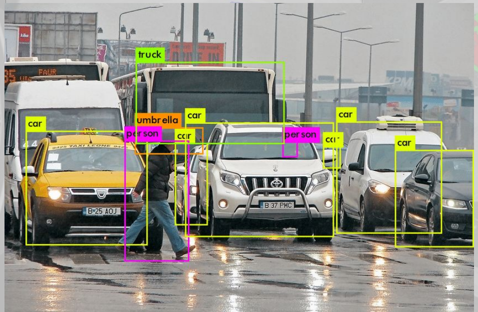
Why PyTorch?

- More research / academia support
- Better customisation ability
- Similar to NumPy
- Dynamism e.g. Dynamic computation graphs

Model Architecture

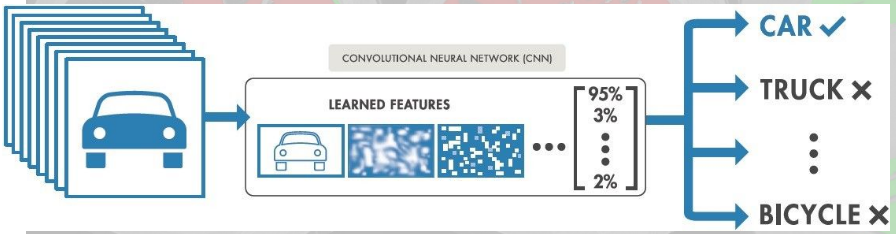
- For multi-label segmentation
- Classification vs. detection vs. segmentation
- Classification: Input image → output class label
- Detection: Input image → output bounding box & class label
- Segmentation: Input image → output image mask

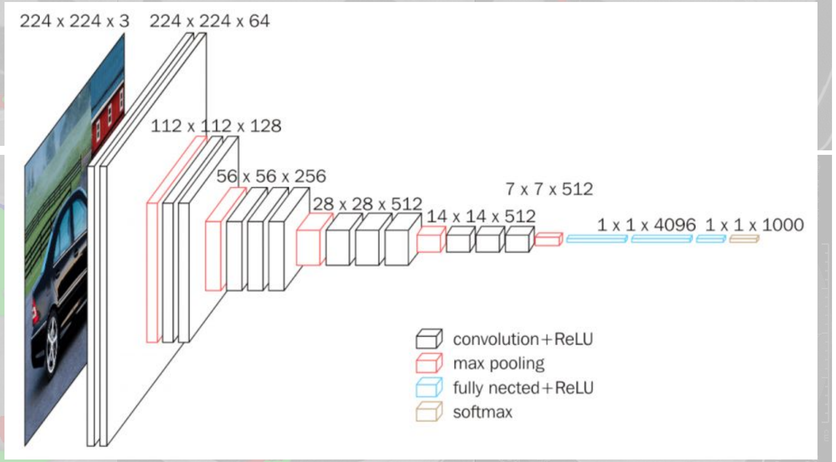




Classification

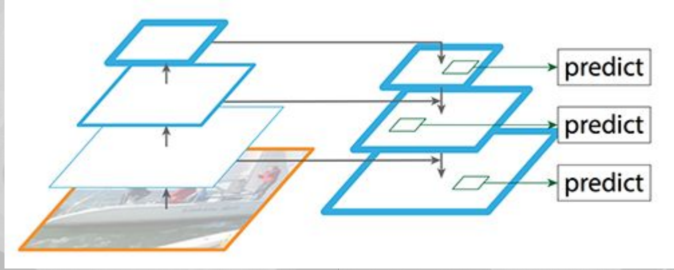
- Input image → output class label (FC layer)
- Can use vanilla Convolutional Neural Networks
- Deep CNNs: accuracy saturation & degradation problem
    - Residual Networks
    - Feature Pyramid Networks





Classification

- ResNets: Shortcut connections
    - Relieves pressure from added deep layers when identity mapping
- FPNs: lateral, top-down connections
    - Fuses feature maps at different scales
    - Each feature map retains local & global information

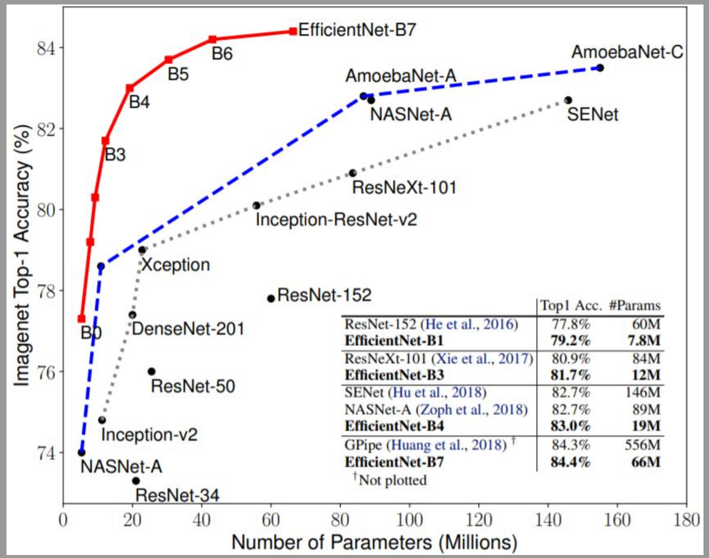


Efficient-Net

- Introduce novel Compound Scaling Method
  - Joint scaling of network 1) depth, 2) width, 3) input resolution

$$
\begin{aligned}
\text{depth: } & d = \alpha^{\phi} \\
\text{width: } & w = \beta^{\phi} \\
\text{resolution: } & r = \gamma^{\phi} \\
\text{s.t. } & \alpha \cdot \beta^{2} \cdot \gamma^{2} \approx 2 \\
& \alpha \geq 1, \beta \geq 1, \gamma \geq 1
\end{aligned}
$$

Upscale computational resources & FLOPS by 2^{ϕ}



SOTA on ImageNet with fewer parameters (less complexity) → more computationally efficient

Segmentation

- Image Segmentation (binary, multi-class), semantic segmentation

Binary Segmentation

Algorithm 1 Binary Segmentation

Result: Binary image mask $M \in \mathbb{R}^{H \times W}$

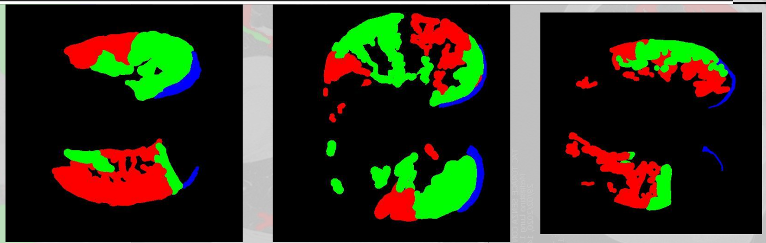
for *pixel in CT_scan_slice* **do**
  if *pixel == infected with COVID-19* **then**
    pixel $\leftarrow 1$
  else
    pixel $\leftarrow 0$
  end
end

Multi-class Segmentation

Algorithm 1 Multi-Class Segmentation

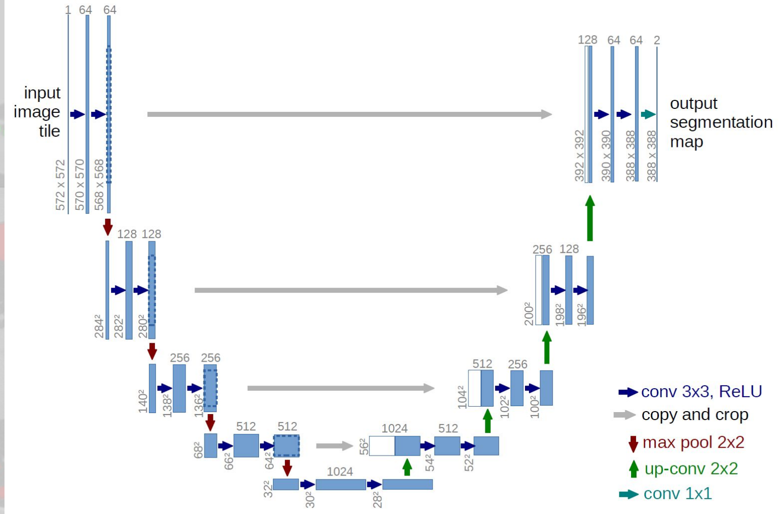
Result: 3D image mask $M \in \mathbb{R}^{3 \times H \times W}$

for *pixel in CT_scan_slice* **do**
  if *pixel == infected with Ground_glass* **then**
    pixel $\leftarrow 1$
  else if *pixel == infected with Consolidation* **then**
    pixel $\leftarrow 2$;
  else if *pixel == infected with Pleural_effusion* **then**
    pixel $\leftarrow 3$;
  else
    pixel $\leftarrow 0$;
  end
end



Fully Convolutional Networks

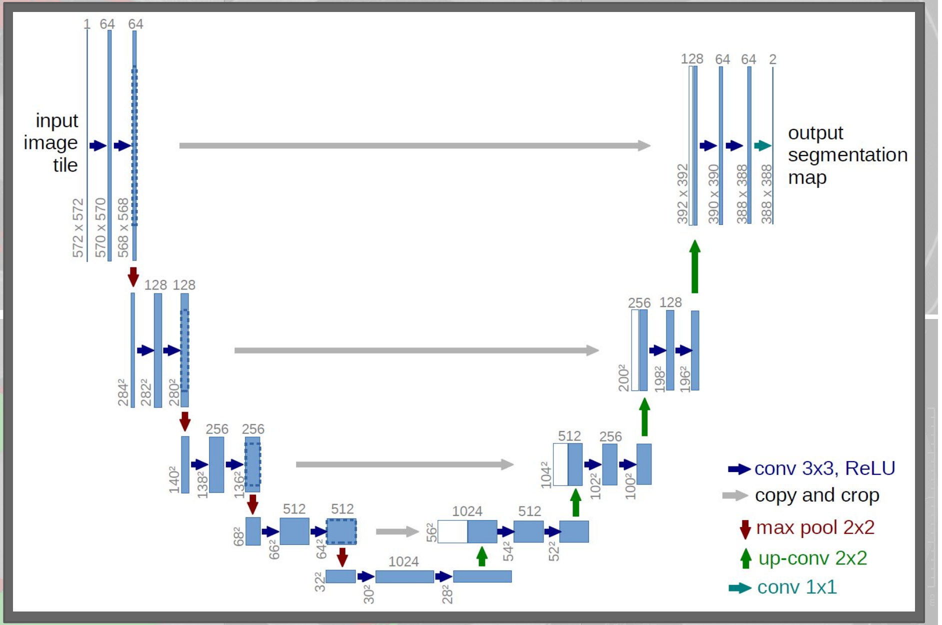
- Used in Corona-Net
- Encoder-decoder network



- Learns convolutional filter directly not function
- U-Net: FCN for biomedical imaging with symmetrical upsampling & downsampling paths, SOTA

U-Net

- Introduce symmetrical contracting & expansive path
- A Fully-Convolutional Network → Computes convolutional filter instead of function
- SOTA in ISBI Challenges
- Tailored to biomedical imaging
- Successful fusion of local to global, spatial-semantic features



Data & Augmentation

- COVID-19 CT segmentation dataset
  - http://medicalsegmentation.com/covid19/
- Augmentation for better generalisation to latent data:
  - Elastic Transformations & Scale Shift ➜ simulate natural deformations of human biological tissue
  - Random cropping ➜ shift invariance
  - Normalisation ➜ grey value invariance
  - Random rotations ➜ rotational invariance

Segmentation Evaluation

Evaluation Metrics	**Accuracies & Losses** (1: binary, 2: multi-class)
1. Dice Coefficient {[0, 1] with 1 best} $$Dice = \frac{2\,\lvert A \cap B \rvert}{\lvert A \rvert + \lvert B \rvert}$$	see table 1
2. Rand Loss {[0, 1] with 0 best} $$RI = \frac{a+d}{\binom{n}{2}},\ RE = 1 - RI$$	see table 2

Dice Coefficient	Rand Loss	Optimiser	Learning Rate
0.5641	0.2167	Adam	1e-02
0.7374	0.1031	Adam	1e-03
0.7965	0.0766	Adam	1e-04
0.4745	0.1591	Adam	1e-05

Dice Coefficient	Rand Loss	Optimiser	Learning Rate
0.5160	0.2490	Adam	1e-02
0.5900	0.2114	Adam	1e-03
0.6160	0.1985	Adam	1e-04
0.5001	0.2565	Adam	1e-05

Future Development

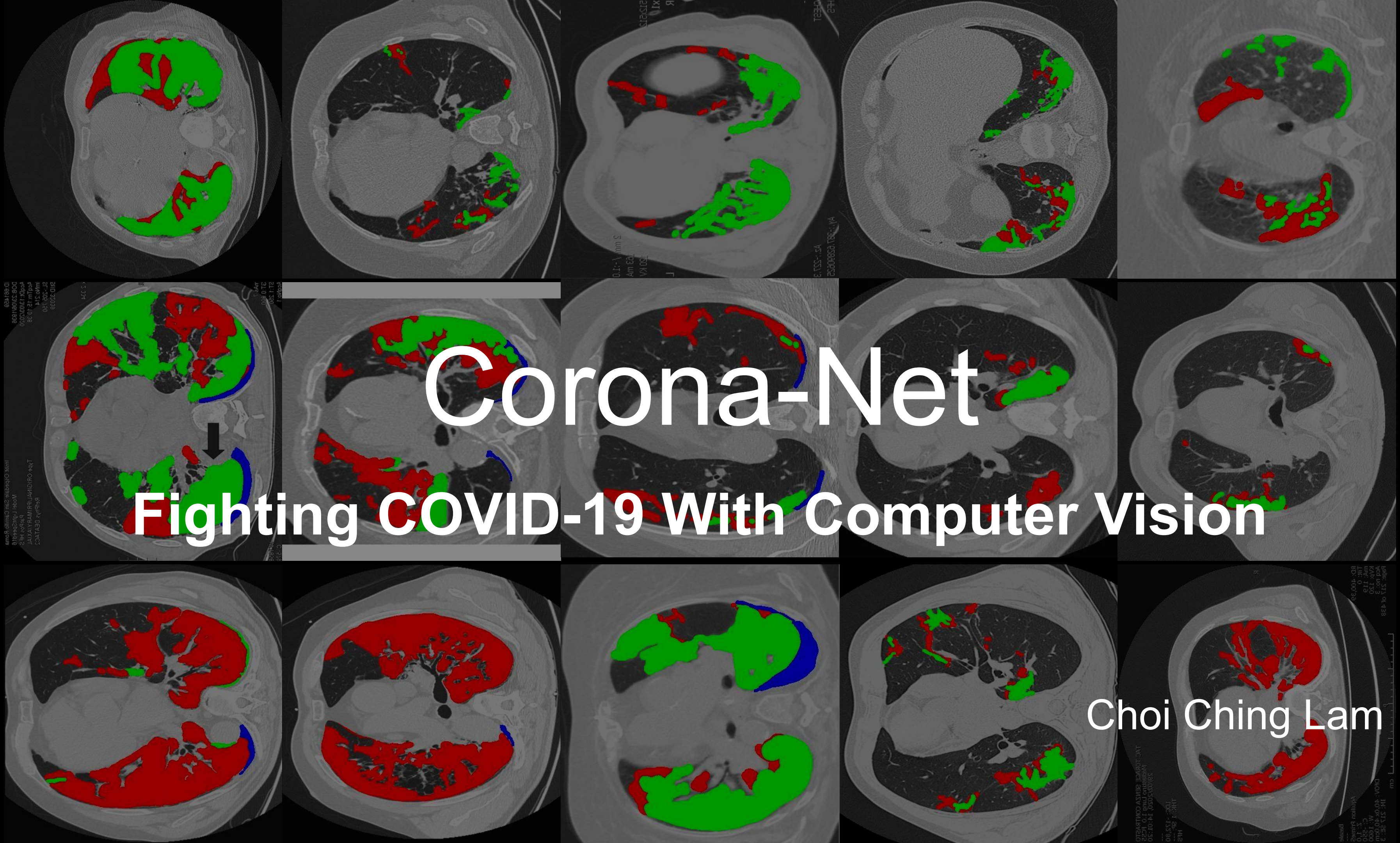
- Recommend Personalised Medicine / Treatment
  - Based on extent (area) and occurrence of particular symptoms of each COVID-19 patient
- Weakly-supervised segmentation
  - Using Global Average Pooling & Object Region Mining
  - No need for labour-intensive mask annotations

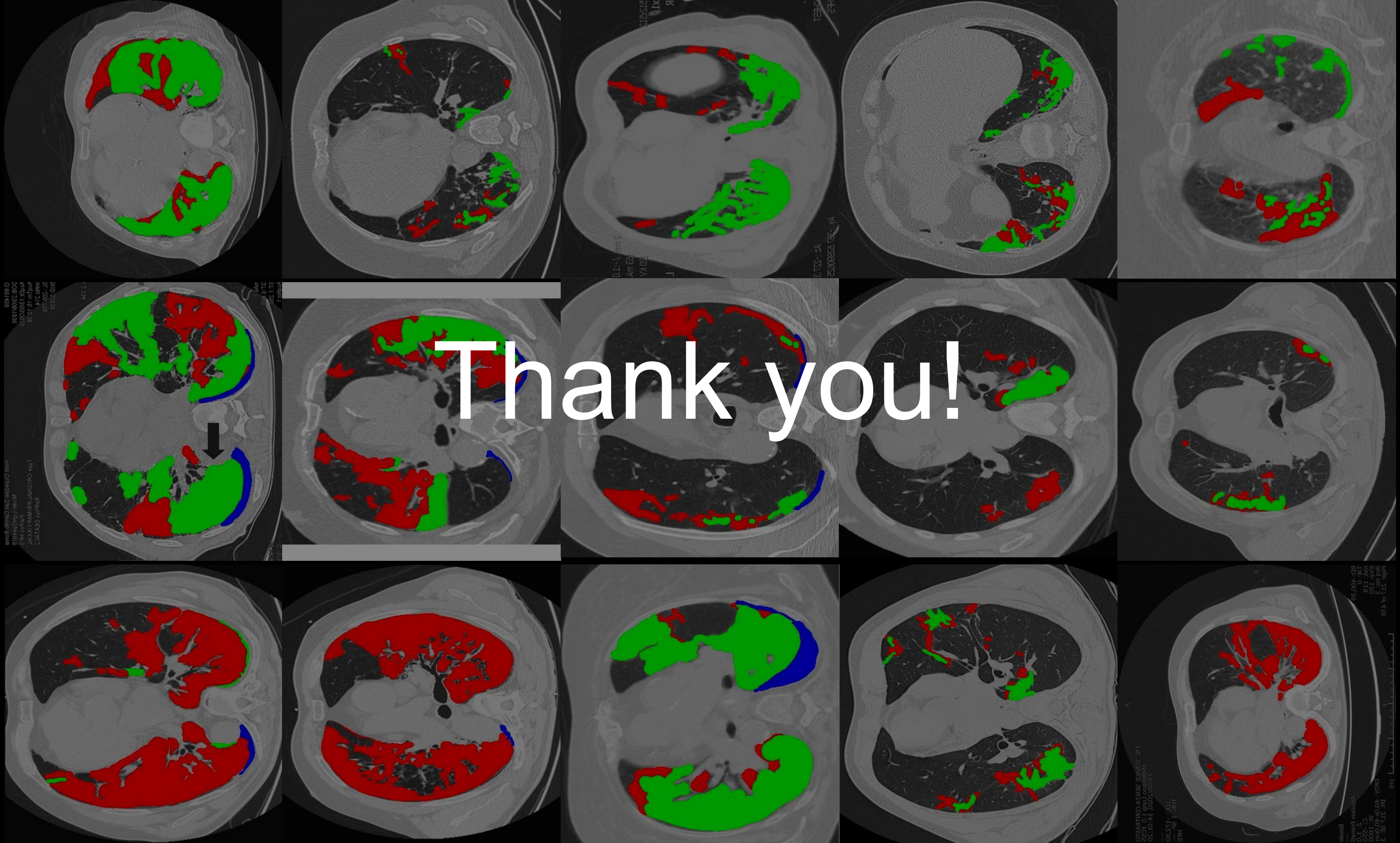
References

- MedicalSegmentation.com. (n.d.). COVID-19 CT segmentation dataset. Retrieved from http://medicalsegmentation.com/covid19/
- K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE CVPR, 2016, pp. 770–778.
- M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," Proceedings of the 36th International Conference on Machine Learning (ICML), 2019.
- Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2117–2125, 2017.
- J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proceedings of IEEE Conference on CVPR, 2015.
- O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In MICCAI. Springer.
- B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. In CVPR, 2016.

Contact Me

- Choi Ching Lam
- Email: ccl5a09@gmail.com
- https://github.com/chinglamchoi
- https://medium.com/@cchoi314
- https://www.linkedin.com/in/ching-lam-choi-7609541a0/
- https://twitter.com/cchoi314


Corona-Net
Fighting COVID-19 With Computer Vision
Choi Ching Lam


Thank you!