



Tuberculosis severity estimation from volumetric CT scans using uniformizing techniques and 3D Convolutional Net

Authors: *Hasib Zunair*, Aimon Rahman and Nabeel Mohammed

Affiliations: North South University, Dhaka 1229, Bangladesh





Overview

- What's this ImageCLEFmed Tuberculosis Severity Scoring Challenge 2019 ([link](#))?
- What we did?
- How we did?
- What results?
- Future works
 - What we did more?
 - How we did?
 - What results?
 -





ImageCLEFmed Tuberculosis.....

Two subtasks were organized in 2019, one was common with the 2018 edition and one new subtask was added:

- *Severity score assessment (SVR subtask).*
- Automatic CT report generation (CTR subtask).
- And some more...





SVR Subtask

The goal of this subtask is to assess the severity based on the Computed Tomography (CT) **image** and additional **clinically relevant meta-data**. The severity score is a cumulative score of severity of a TB case assigned by a medical doctor. Originally, the score varied from 1 ("critical/very bad") to 5 ("very good").

The original severity score was included as training metadata **but** the final score that participants had to assess was reduced to a binary category: "LOW" (scores 4 and 5) and "HIGH" (scores 1, 2 and 3). Hence, the task is reduced to a **binary classification problem**.





Data Sets

Consists of **335 chest CT scans** of TB patients along with a set of clinically relevant meta-data, divided into 218 patients for **training** and 117 for **testing** (labels hidden, obviously!!).

The selected meta-data include the following binary measures: disability, relapse, symptoms of TB, comorbidity, bacillary, drug resistance, higher education, ex-prisoner, alcoholic, smoking, and severity.

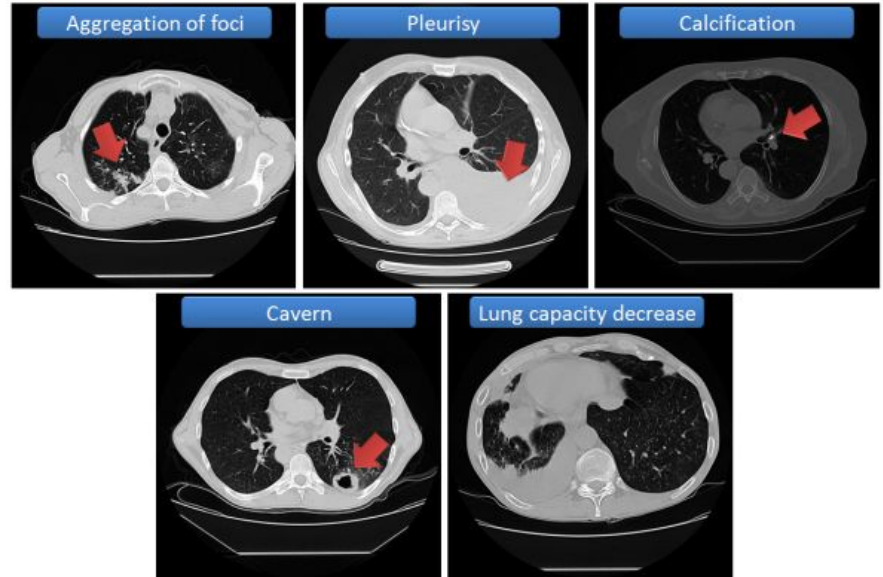
Severity(Target label for the given CT image): 0 or 1



Data Sets continued..

- 3D CT images with a slice size of 512×512 pixels and number of slices varying from about **50** to **400** were provided.
- CT images stored in NIFTI file format with .nii.gz file extension (g-zipped .nii files).
- Format stores raw voxel intensities in Hounsfield units (HU).

Symptoms of TB



https://www.researchgate.net/publication/334207545_Overview_of_ImageCLEFt_ubercolosis_2019_-_Automatic_CT-based_Report_Generation_and_Tuberculosis_Severity_Assessment

Show CT Image GIF





What and how we did?

The mighty slice selection technique (SS)

We use **only** the 3D CT images and **ground truth**, excluded the meta-data.

Resize individual *slices* of the 3D input volume to 128×128 .

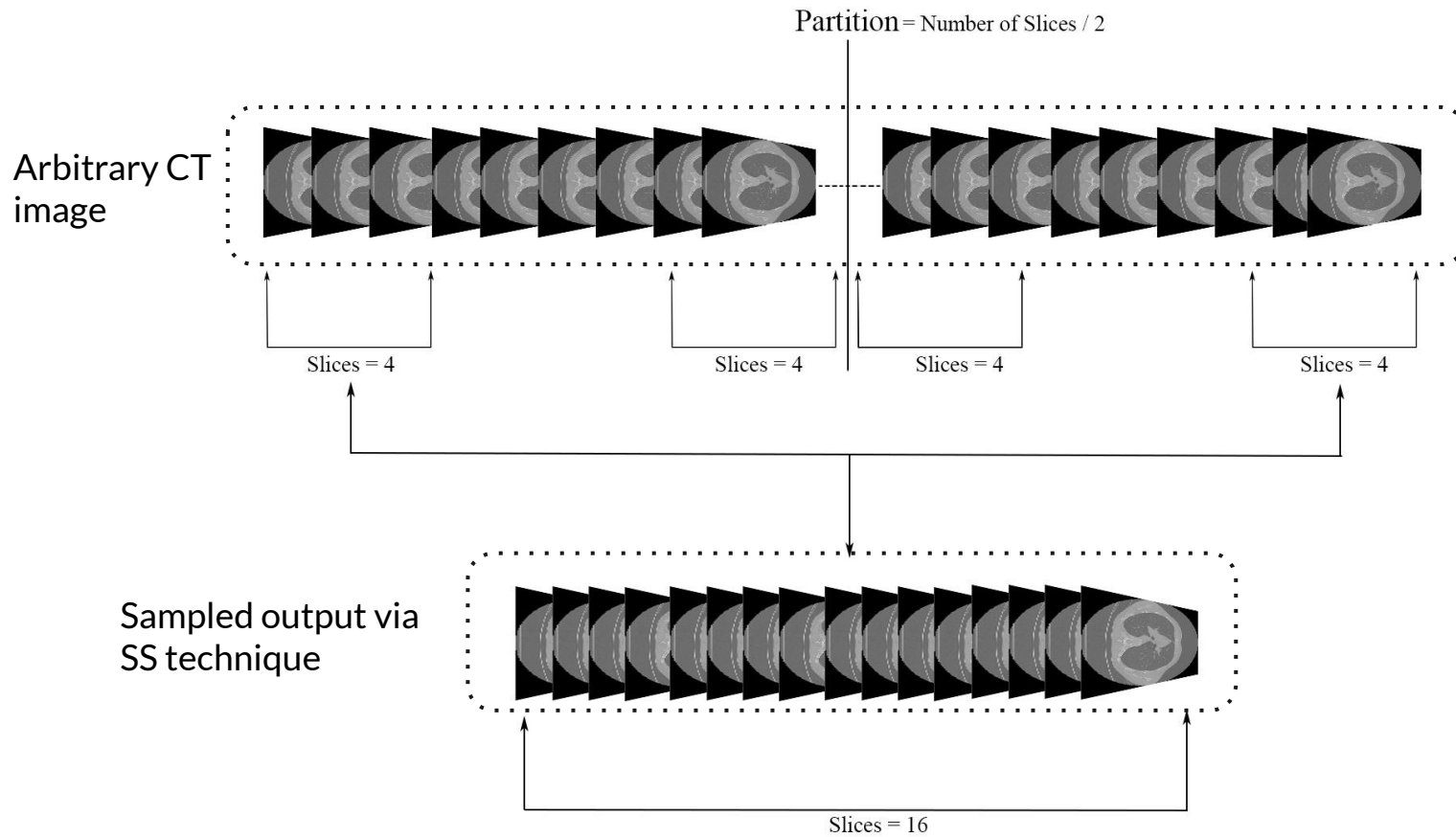
Given an input volume: i) the first 4 slices, ii) the middle 8 slices and iii) the last 4 slices are sampled. (The middle slice of the input volume is obtained by taking the half of the input volume depth.)

These three sub components are then *stacked depthwise* to reconstruct the desired input volume

Motivation: **eliminates the problem of GPU exhaustion during optimization**

****we know this has huge limitations, don't throw tomatoes just yet...**

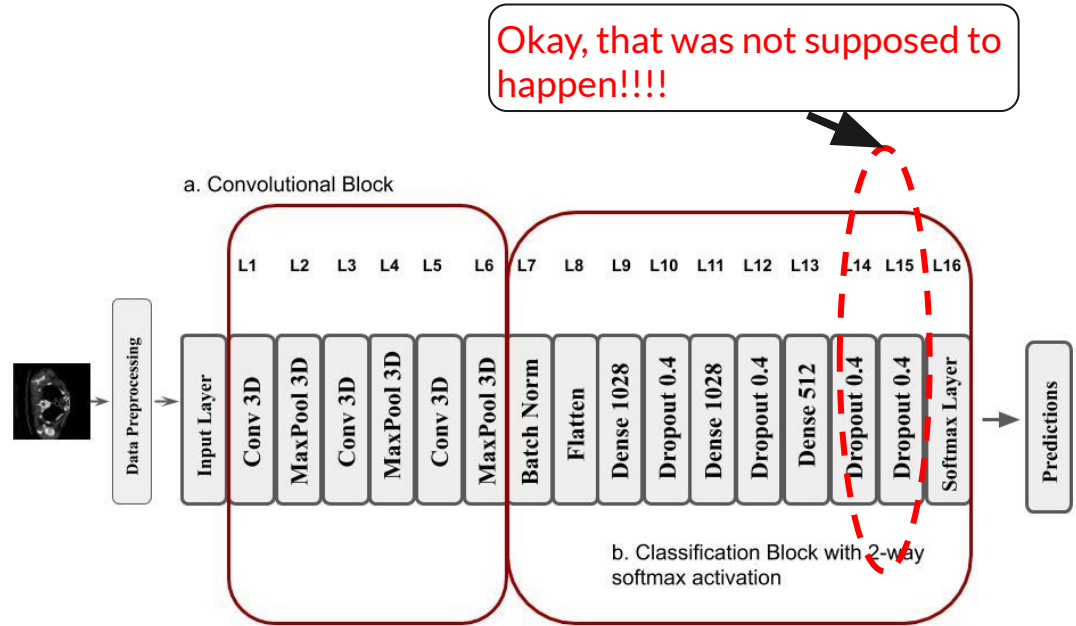




https://www.researchgate.net/publication/334680379_Estimating_Severity_from_CT_Scans_of_Tuberculosis_Patients_using_3D_Convolutional_Nets_and_Slice_Selection/citations

Our 3D ConvNet

- Input shape $128 \times 128 \times \text{depth}$
- Filters for conv (32, 32, 32)
- Filter size $2 \times 2 \times 2$
- 2-way softmax activation for final probability predictions
- 23,808,378 parameters



https://www.researchgate.net/publication/334680379_Estimating_Severity_from_CT_Scans_of_Tuberculosis_Patients_using_3D_Convolutional_Nets_and_Slice_Selection/citations



How we trained our network

- Random initialization
- Stochastic Gradient Descent (SGD) optimizer with learning rate $1e-4$
- Binary cross entropy (BCE) loss
- All layers trained for 300 epochs

Name	Input Volume Depth Size	Batch Size
CFG-A	32	4
CFG-B	16	16

https://www.researchgate.net/publication/334680379_Estimating_Severity_from_CT_Scans_of_Tuberculosis_Patients_using_3D_Convolutional_Nets_and_Slice_Selection/citations



What results?

Table 2. Performance on final test set

Name	Area Under ROC Curve	Test Set Accuracy	Best Validation Accuracy	Test-Val Accuracy Margin
CFG-A	0.611	61.5	68	6.5
CFG-B	0.57	53.8	82.5	28.7



https://www.researchgate.net/publication/334680379_Estimating_Severity_from_CT_Scans_of_Tuberculosis_Patients_using_3D_Convolutional_Nets_and_Slice_Selection/citations

Compared to others?

Table 4. Detailed results obtained in the SVR task by the best run of each group.

Group name	AUC	Accuracy	Kappa
UIIP_BioMed	0.7877	0.7179	0.4310
UIIP	0.7754	0.7179	0.4321
HHU	0.7695	0.6923	0.3862
CompElecEngCU	0.7629	0.6581	0.3289
SD VA HCS/UCSD	0.7214	0.6838	0.3646
MedGIFT	0.7196	0.6410	0.2720
UniversityAlicante	0.7013	0.7009	0.4014
MostaganemFSEI	0.6510	0.6154	0.2335
SSN CoE	0.6264	0.6068	0.2109
UoAP	0.6111	0.6154	0.2272
FIIAugt	0.5692	0.5556	0.1005





Resources

- Paper:
[https://www.researchgate.net/publication/334207545 Overview of ImageCLEFtuberculosis 2019 - Automatic CT-based Report Generation and Tuberculosis Severity Assessment](https://www.researchgate.net/publication/334207545)
- GitHub: <https://github.com/hasibzunair/tuberculosis-severity>
- Published in CLEF 2019 Working Notes
<http://www.dei.unipd.it/~ferro/CLEF-WN-Drafts/CLEF2019/>



Anyone tired and need a break?

Questions?



Future works in present

- Improve previous 3D Convolutional Network (less parameters, better performance)
- Find loopholes in previous data processing technique and improve upon them



What we did next?

Subset Slice Selection (SSS) to Even Slice Selection (ESS)

Even slice selection: Now, we improve upon this method by sampling **evenly** such that the depth information is preserved, to an extent.

We take a target depth N and a scan depth of size D . A spacing factor is then determined which is governed by the following :

$$S = \frac{D}{N}$$

We sample slices maintaining the spacing S throughout the entire volume. This gives us better generalized representation of the chest 3D scan compared to previously *subset slice selection* (SSS) technique.





Spline Interpolated Zoom (SIZ)

Since the concept of *slices* result in mitigating the volumetric information, we refrain from this.

In this technique, instead of manually selecting a subset of slices, we pre-determine a constant target depth size N . We then take each volume, i) calculate its depth D ii) zoom it along z-axis/depth-wise by factor shown in Eq1. and by using spline interpolation [1] of order 3(x, y, z). Here, the input volume is extended by replicating the nearest pixel. The zooming method does not change the semantic meaning of the slices.

$$\text{Depth factor: } \frac{1}{D/N}$$

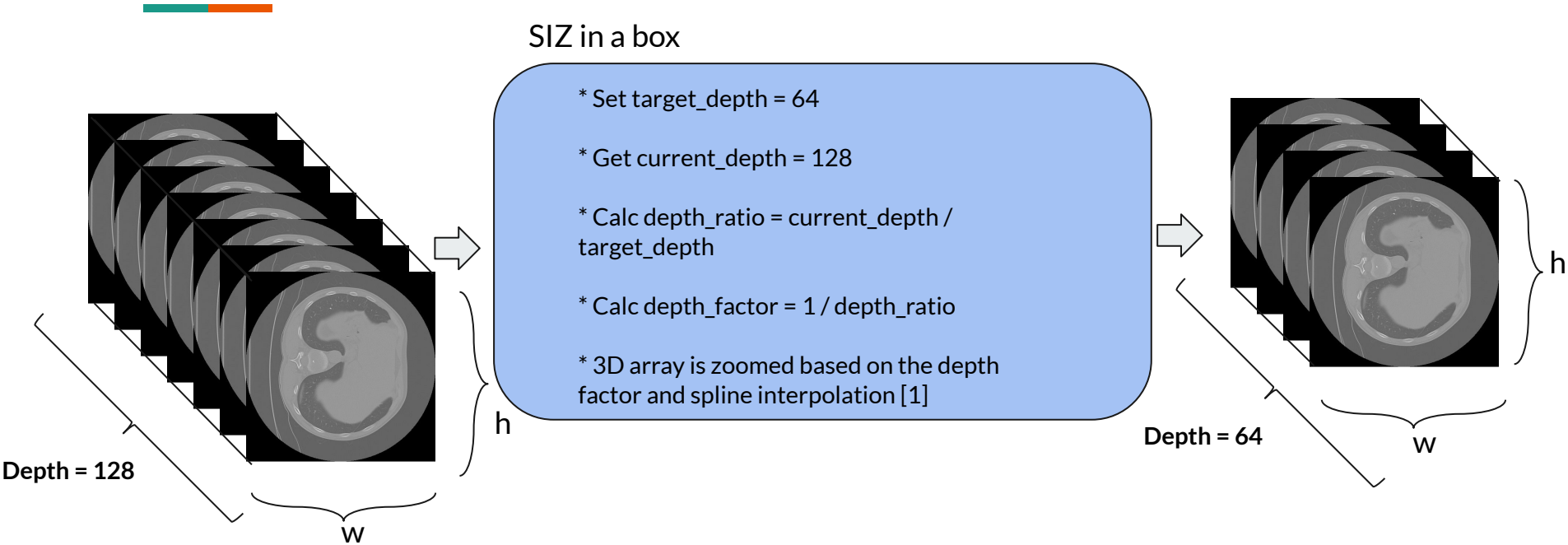
[1] De Boor, Carl. "Bicubic spline interpolation." Journal of mathematics and physics 41, no. 1-4 (1962): 212-218



****These are NOT slices, the entire volume is processed as a whole.****

SIZ in a box

- * Set target_depth = 64
- * Get current_depth = 128
- * Calc depth_ratio = current_depth / target_depth
- * Calc depth_factor = 1 / depth_ratio
- * 3D array is zoomed based on the depth factor and spline interpolation [1]



[1] De Boor, Carl. "Bicubic spline interpolation." Journal of mathematics and physics 41, no. 1-4 (1962): 212-218

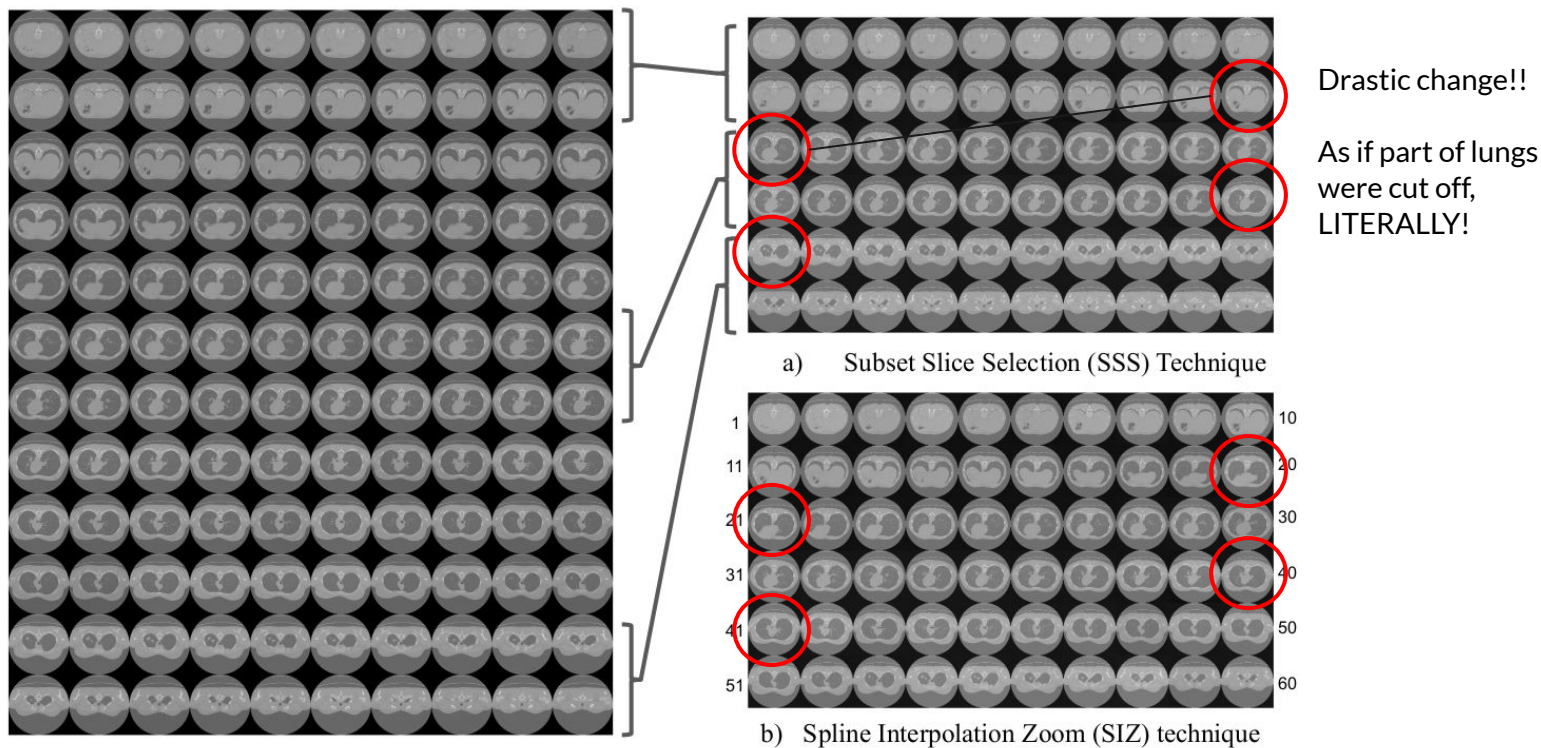


Fig. 2. Qualitative comparison of a) SSS and b) SIZ. It is evident that using SSS prevents the depth information to be utilized and changes the semantic meaning of the volumetric data as a subset of the *slices* are being deliberately discarded. Hence, resulting in information loss where SIZ on the other hand attains.



Data preprocessing

Normalization: The slices initially have **raw voxel intensities in Hounsfield units (HU)**, where maximum values are 16816.95 and minimum values -8611.168, we normalize them between 0 to 1.

$$X = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2)$$

where X is the raw data matrix, X_{min} is the minimum value in the tensor, X_{max} is the maximum and finally X being the normalized tensor.





Data preprocessing and beyond...

Zero centering: The slices are zero centered, which means the mean value of the slices are zero, by subtracting pixel mean from all pixels.

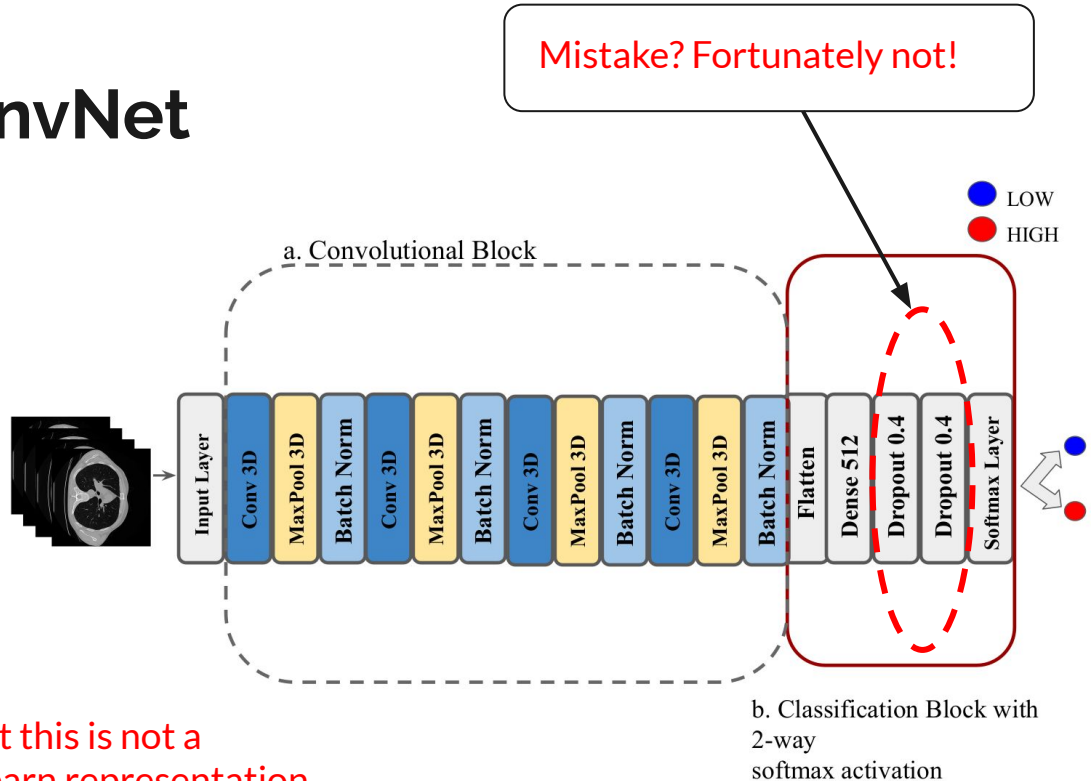
$$\hat{X} = X - X_{mean} \quad (3)$$

where X is the input tensor, X_{mean} is the mean value of the tensor and \hat{X} is the zero centered tensor.



Our scaled 3D ConvNet

- Input shape $128 \times 128 \times 64$
- Convolutional block consists of *conv-maxpool-bn* modules
- Filters for conv (64, 64, 128, 256)
- Filter size $3 \times 3 \times 3$
- 2-way softmax activation for final probability predictions
- **10,658,498** parameters (previously 23,808,378)



Patchwise training considered, but this is not a segmentation problem. Need to learn representation as a whole! (Need to know **label** for each patch)



How we trained our network

- Random initialization
- Stochastic Gradient Descent (SGD) optimizer with learning rate $1e-4$
- Binary cross entropy (BCE) loss
- All layers trained for **100** epochs (previously 300)



What results?

Table 1. Performance measures compared with our previous results which are evaluated on the final test set provided by ImageCLEF Tuberculosis Severity Assessment 2019 consisting of 117 chest CT scans.

Previous work ->>

CFG	Data Pro- cessing	Data Nor- malization	Zero Cen- tering	AUC Score	Accuracy Score
clef-cfg-a	SSS(CFG-A)	No	No	0.611	0.615
clef-cfg-b	SSS(CFG-B)	No	No	0.57	0.538
sss-d0-z0	SSS	No	No	0.626	0.538
sss-d1-z0	SSS	Yes	No	0.635	0.573
sss-d1-z1	SSS	Yes	Yes	0.64	0.598
ess-d0-z0	ESS	No	No	0.639	0.607
ess-d1-z0	ESS	Yes	No	0.667	0.598
ess-d1-z1	ESS	Yes	Yes	0.67	0.611
siz-d0-z0	SIZ	No	No	0.648	0.581
siz-d1-z0	SIZ	Yes	No	0.652	0.607
siz-d1-z1	SIZ	Yes	Yes	0.73	0.675



Squeezed our way in 5th, post challenge!!

Table 2: Performance metric results compared with previous top ranking approaches on ImageCLEF benchmark. The results reported on each of the metrics are on the Imageclef final test set which consists of 117 CT images. Boldface indicates our proposed approach

Group Name	AUC	ACC
UIIP_BioMed [31]	0.7877	0.7179
UIIP [32]	0.7754	0.7179
HHU [43]	0.7695	0.6923
CompElecEngCU [37]	0.7629	0.6581
siz-d1-z1 (Ours)	0.7300	.6750
SD VA HCS/UCSD [33]	0.7214	0.6838
MedGIFT [23]	0.7196	0.6410
UniversityAlicante [38]	0.7013	0.7009
MostaganemFSEI [24]	0.6510	0.6154
SSN CoE [41]	0.6264	0.6068
UoAP (Ours: [26])	0.6111	0.6154
FIIAugt [42]	0.5692	0.5556



Approaches without meta-data?

Table 3: Performance metric results on the ImageCLEF online benchmark of the proposed top ranking methods which only used the 3D CT Image data.. Boldfaces indicate our proposed approach

Group Name	AUC	ACC
siz-d1-z1 (Ours)	0.7300	.6750
MedGIFT [23]	0.7196	0.6410
UniversityAlicante [38]	0.7013	0.7009
MostaganemFSEI [24]	0.6510	0.6154
SSN CoE [41]	0.6264	0.6068
UoAP (Ours: [26])	0.6111	0.6154
FIIAugt [42]	0.5692	0.5556



Limitations of this work?

Unable to conduct both **success** and **failure** case analysis as the test set labels are hidden.

Interpretability of the approach.





Conclusion

Data processing pipeline to preserve depth information and **discard the concept of *slices***

3D Convolutional Network for volumetric image classification (half the number of trainable parameters and epochs) employed in Computed Tomography (CT) images

Ablation of experiments

Goal is not to claim the state-of-the-art but rather introduce a data processing pipeline that can be integrated into existing convolutional architectures to address volumetric image analysis and refrain from using the concept of *slices*.



More future works?

- Scale 3D ConvNet inspired by work from EfficientNet[1].
- 3D data augmentation
- Leverage models pretrained on 3D datasets both in natural and medical image setting.



****Extended work was rejected from ECIR 2020****

[1] Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv preprint arXiv:1905.11946*.



PRESENTATION FINISHED



NOW ANY QUESTIONS?

makeameme.org

Feedback!!

