

# ViPTT-Net: Video pretraining of spatio-temporal model for tuberculosis type classification from chest CT scans

Hasib Zunair<sup>1</sup>, Aimon Rahman<sup>2</sup> and Nabeel Mohammed<sup>2</sup>

<sup>1</sup>Concordia University, Montreal, QC, Canada

<sup>2</sup>North South University, Dhaka, Bangladesh

## Abstract

Pretraining has sparked groundswell of interest in deep learning workflows to learn from limited data and improve generalization. While this is common for 2D image classification tasks, its application to 3D medical imaging tasks like chest CT interpretation is limited. We explore the idea of whether pretraining a model on realistic videos could improve performance rather than training the model from scratch, intended for tuberculosis type classification from chest CT scans. To incorporate both spatial and temporal features, we develop a hybrid convolutional neural network (CNN) and recurrent neural network (RNN) model, where the features are extracted from each axial slice of the CT scan by a CNN, these *sequences of image features* are input to a RNN for classification of the CT scan. Our model termed as ViPTT-Net, was trained on over 1300 video clips with labels of human activities, and then fine-tuned on chest CT scans with labels of tuberculosis type. We find that pretraining the model on videos lead to better representations and significantly improved model validation performance from a kappa score of 0.17 to 0.35, especially for under-represented class samples. Our best method achieved 2nd place in the ImageCLEF 2021 Tuberculosis - TBT classification task with a kappa score of 0.20 on the final test set with only image information (without using clinical meta-data). All codes and models are made available.<sup>1</sup>

## Keywords

Tuberculosis, 3D image classification, Human Action Recognition, Spatial-Temporal Information, Pre-training

## 1. Introduction

Tuberculosis (TB) is a potentially fatal disease that generally affects the lungs. The disease spreads through cough, spit or sneeze and can remain latent within the human body. Although X-rays and microscopic analysis of bodily fluid are generally used to diagnose the disease, Computed Tomography (CT) provides detailed information about the infection. Deep learning models demonstrate promising results in diagnosing TB from both X-rays and CT scans [1, 2, 3, 4]. These methods are also proven to be effective for severity scoring of the infection as well [5, 6, 7]. However, the types of TB can vary and that may require a different course of treatment, making

<sup>1</sup><https://github.com/hasibzunair/viptt-net>

CLEF 2021 – Conference and Labs of the Evaluation Forum, September 21–24, 2021, Bucharest, Romania

✉ hasibzunair@gmail.com (H. Zunair); aimon.rahman@northsouth.edu (A. Rahman);

nabeel.mohammed@northsouth.edu (N. Mohammed)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

the identification of TB type an important real-life problem. The deep learning models are yet to provide high accuracy in this particular task.

The most challenging part of working with CT scan data is, it is three-dimensional (3D), meaning along with height and width, each data point contains depth information. Processing 3D data can be very computationally expensive and may require pre-processing before training a model. These pre-processing steps may include choosing selective slices or transforming slices into uniform sizes. Both of these techniques may contribute to losing some depth information. One of the most effective method of processing 3D data is to resize it into fixed dimension using spline interpolation along x, y and z- axis which was demonstrated in [5]. Although the method showed promising result in TB severity scoring, might not perform well on other more sophisticated tasks such as classifying TB types.

An alternative to directly processing 3D data is to decompose each scan into individual slices and afterward feeding them 2D CNN [8, 9, 10, 11]. The problem with this method is, as the slices are considered independent to each other, the spatial information along the z-axis is missing during training. Moreover, in this method, the label for the whole volume is assigned to each individual slice, which might not be the case when it comes to CT scans. More specifically, a CT scan of infected lungs may contain uninfected 2D slices.

In this work, we build a model to predict multiple tuberculosis types from chest CT scans. We pretrain a hybrid convolutional neural network (CNN) and recurrent neural network (RNN) model on human action recognition task, and then fine-tune the model for the tuberculosis type classification task. Pretraining significantly improves performance, especially for minority classes. The method is evaluated on the Image-CLEF 2021 Tuberculosis - TBT classification task which achieves 2nd place <sup>1</sup> with a kappa score of 0.20 and accuracy of 0.42.

We summarize our contributions as follows:

1. We pretrain a hybrid CNN-RNN model, termed ViPTT-Net, on human action recognition task, and fine-tune the model on a small dataset of CT scans with labels indicating tuberculosis types.
2. We show pretraining ViPTT-Net on realistic videos improve performance for tuberculosis type classification, especially for under-represented class samples.
3. We evaluate our best method on the Image-CLEF 2021 Tuberculosis - TBT classification task which achieves 2nd place overall.

## 2. Methodology

We start by problem formulation followed by presenting the main building blocks of our proposed method for multi-class CT image classification.

### 2.1. Problem Formulation

Given the labels of a set of CT scans, the objective is to predict the unknown labels of the new CT scans <sup>2</sup>. More specifically, our goal is to learn a discriminative function  $f(\mathbf{X}) \in \{1, 2, 3, 4, 5\}$ ,

---

<sup>1</sup>[imageclef-2021-tuberculosis-tbt-classification/leaderboards](https://www.imageclef.org/2021/medical/tuberculosis)

<sup>2</sup><https://www.imageclef.org/2021/medical/tuberculosis>

where the numbers represent the tuberculosis type: *Infiltrative*, *Focal*, *Tuberculoma*, *Miliary* and *Fibro-cavernous* respectively.  $\mathbf{X}$  represents a CT scan volume of size  $D \times W \times H$ , where  $D$ ,  $W$ , and  $H$  represent the depth, width, and height of the volume respectively. The task is considered as a multi-class volumetric image classification problem. A 3D volumetric scan  $\mathbf{X}$  can also be viewed as a time-series of 2D slices  $\{X_1, \dots, X_D\}$ . Therefore, we can also frame the task as a time-series sequence classification problem.

## 2.2. ViPTT-Net Model

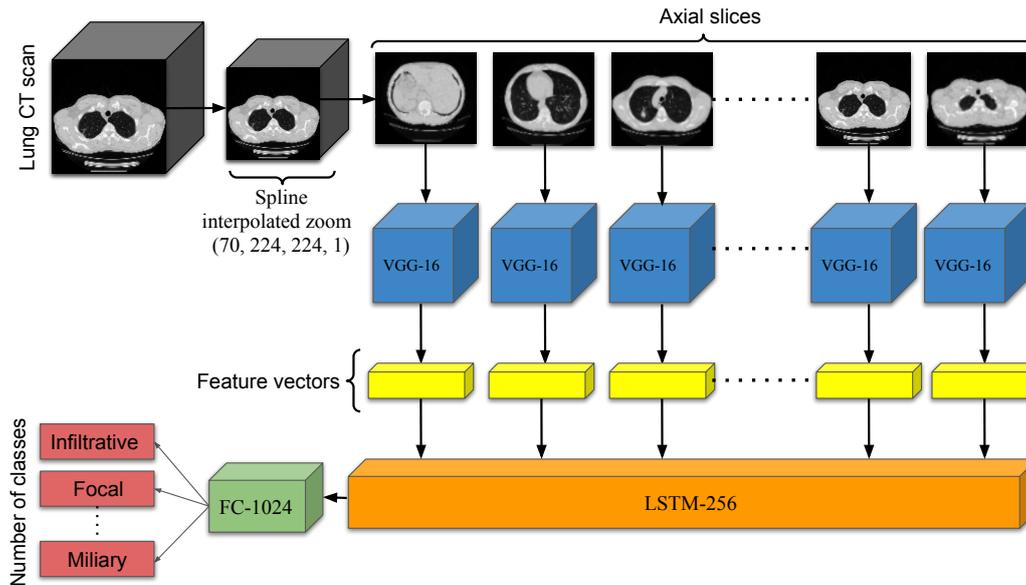
While convolutional neural networks (CNNs) have shown promising results at processing image data [12, 13, 14], the same can be said for recurrent neural networks (RNN) for sequential data [15, 16, 17]. We developed a hybrid convolutional neural network (CNN) and recurrent neural network (RNN) model termed ViPTT-Net which is capable of incorporating both spatial and temporal features in the learning process. ViPTT-Net takes as input a CT scan and outputs probability predictions which indicates the type of tuberculosis in that CT scan. To deal with arbitrary volumes, we first resize the CT scan to a fixed size  $70 \times 224 \times 224$  using spline interpolated zoom (SIZ) [5] which exploits the full geometry of the 3D volume by interpolating over the  $z$ -axis.

**Learning spatial features using CNN.** To learn spatial features, ViPTT-Net consists of a VGG-16 model as the feature extractor which is pretrained on ImageNet [18]. The VGG-16 model has 16 layers with learnable weights: 13 convolutional layers, and 3 fully connected layers [19]. We extract features from the last convolutional layer which results in a 512 dimensional feature vector for a 2D input image. Since a CT scan consists of multiple 2D axial slices (70 in our case), we wrap the VGG-16 model in a time-distributed layer which is then applied to every axial slice (temporal frame) of the CT scan independently. The final output is a *sequence of image features* where the sequence length is 70 and each of the sequences are a 512 dimensional feature vector.

As the VGG-16 feature extractor accepts inputs of 3-channels, we map the 1-channel axial slices of the CT scan slices to 3-channel using a convolutional layer with 3 filters and kernel size of  $1 \times 1 \times 1$  before input to the feature extractor. We use this instead of converting to pseudo-color image by just duplicating intensity to all channels to mimic 3-channel because it become more computationally expensive to train.

**Learning temporal features using RNN.** Following the above, the sequence of image features are aggregated using an RNN. RNNs are a type of neural network which transforms a sequence of inputs into a sequence of outputs. Using an RNN, the temporal order of the axial slices is preserved. We use the long short-term memory (LSTM) [20] with 256 units and tanh activation. It is important to mention that since we are interested in classifying the full sequence, the output from the LSTM results in a 256 dimensional aggregated feature vector from the last time step coming from the full sequence of image features, rather than producing sequences of outputs. We also experimented with gated recurrent unit (GRU) but did not get good results on the validation set.

Finally, the aggregated 256 dimensional feature vector is passed to a dense layer with 1024 units with rectified linear hidden units (ReLU) activation and a dense layer with a softmax function (i.e. a dense softmax layer of 5 units for the multi-class classification case) that yields



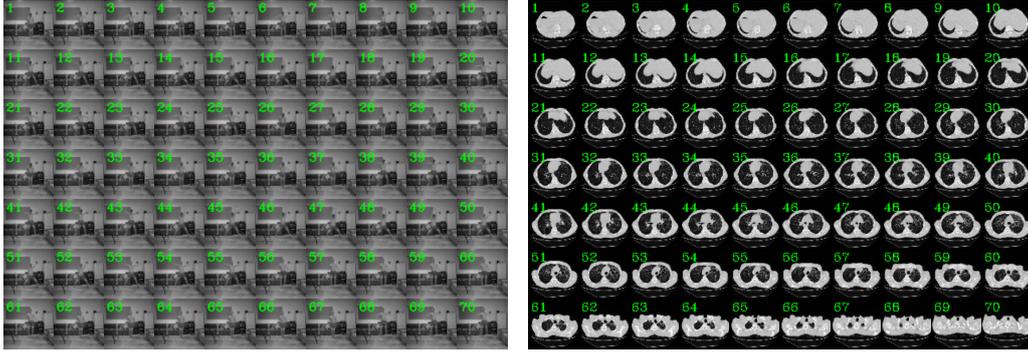
**Figure 1:** Schematic layout of the hybrid CNN-RNN model ViPTT-Net. Given a 3D CT scan of arbitrary size, uniform resizing is performed across all dimensions using SIZ [5]. Features are extracted from all the axial slices of the processed CT scan to output a sequence of image features using a VGG-16 model. These sequence of image features are input to an LSTM layer followed by dense layers of 1024 neurons and finally 5 with softmax activation for the multi-class classification problem.

the probabilities of predicted classes. While training on UCF50, instead of a softmax with 5 units, we use a softmax with 10 units for the 10 classes. An illustration of ViPTT-Net is shown in Fig 1.

### 2.3. Pretraining on Human Action Recognition Task

ViPTT-Net was first pretrained on a subset of the UCF50 dataset [21]. UCF50 is human action recognition dataset with 50 action categories, consisting of realistic videos taken from youtube. The classes of different action are Baseball Pitch, Jumping Jack, Kayaking *etc.* Due to compute constraints, we use 1366 videos from 10 classes selected randomly: Mixing, Tennis Swing, Horse Riding, Jump Rope, Jumping Jack, Baseball Pitch, Rowing, SkateBoarding, Walking With Dog, Skijet. Video clips are resized to  $70 \times 224 \times 224$ , where 70 is the number of temporal frames (image sequences), and 224 is width and height of the temporal frame. It is also important to mention that each RGB frame of the video is converted to grayscale (single channel) since the CT scans also have single channel values. An instance of an action video is shown in Fig 2a.

Similar to pretraining for 2D image problems [22, 23], after training ViPTT-Net on a subset of UCF50 dataset, the network was fine-tuned on the tuberculosis type classification task by replacing the final fully connected 10-way softmax layer with a 5-way softmax. During training ViPTT-Net on UCF50, the weights of the VGG-16 feature extractor was frozen. And while fine-tuning on the tuberculosis type classification task, all the network layers were trained.



(a) Sequence of images from a video of a man playing *Jumping Jack*. (b) Sequence of axial images from a CT scan with the label *Infiltrative*.

**Figure 2:** Illustration of image sequences of a video and a CT scan from the UCF50 and ImageCLEF 2021 - TBT datasets. Both samples are resized to depth of 70.

## 2.4. Weighted Loss and Data Augmentation

Due to heavy class imbalance, we assign weights in the loss function for each tuberculosis type with the goal to reduce biasness towards the over-represented class samples. Prior to training using weighted loss, the weights are computed over the training set.

It is a standard practice to perform data augmentation to improve generalization, especially when there are limited number of training data. Data augmentation basically creates modified versions of the input data in a dataset through random transformations such as horizontal and vertical flip, zoom augmentation, horizontal and vertical shift, *etc.* While training, the 3D CT scans are rotated with degree of rotations picked randomly from  $[-20, -10, -5, 0, 5, 10, 20]$  as a form of data augmentation. We use a wider range to cover a larger distribution of augmented images. Notice that we added 0 in the range which means that the model looks at both augmented and non-augmented data. We experimented without adding 0 in the range and led to poor results on the validation set. We also tried blur and random shifts but did not get good results.

## 3. Experiment Setup

We describe experimental details, *i.e.*, dataset, implementation details, evaluation metrics, *etc.*, and present quantitative results, comparing different training strategies using ViPTT-Net.

### 3.1. Datasets and Preprocessing

The dataset is provided by ImageCLEF Tuberculosis Type 2021 Challenge [24, 25]. It contains CT scans of a total of 1338 TB patients, 917 of them have been used for training and 421 for the test set. Each scan has a label that indicates one of the TB types - Infiltrative, Focal, Tuberculoma, Miliary and Fibro-cavernous. Each CT scan belongs to only one patient. The images have dimensions of 512 x 512 pixels with varying depth sizes. In addition to labels, some scans have additional meta-data. All the scans have auto-generated lungs mask, although some of these

masks are missing in largely affected areas or have rough bounds, majority of the scans are quite accurate. The original data is in NIFTI format, storing the raw voxel intensity in Hounsfield units (HU). An instance of a CT scan with Infiltrative tuberculosis type is shown in Fig 2b.

### 3.2. Implementation Details

All experiments are performed on a Linux workstation running 4.8Hz and 64GB RAM with and RTX 3080 GPU. Experiments are conducted using Python programming language [26]. ViPTT-Net is implemented in Keras [27], with TensorFlow backend [28].

ViPTT-Net is trained end-to-end using stochastic gradient descent (SGD) optimization algorithm to minimize the categorical cross-entropy loss function with an initial learning rate of 0.001 and batch size of 2. A factor of 0.1 is used to reduce the learning rate once the loss stagnates. Training is continued until the validation loss stagnates using an early stopping mechanism, and then the best weights are retained. To keep data proportions same and ensure reproducibility, we perform a stratified train and validation split with a ratio of 80/20 (732/184 CT scans) on the training data provided by ImageCLEF 2021 - TBT.

Similar steps are followed while training on a subset of the UCF50 dataset, except we do not use any data augmentation in this case. Also, the pretrained VGG-16 feature extractor was kept frozen during training since both ImageNet and UCF50 samples are natural images/videos.

### 3.3. Evaluation Metrics

According to challenge rules, the task is evaluated as a multi-class classification problem. The main evaluation metric is the kappa score. Kappa measures the inter-rater reliability for categorical items.

$$\kappa \equiv \frac{p_0 - p_e}{1 - p_e} \quad (1)$$

Here,  $p_0$  indicates the relative observed agreement among raters and  $p_e$  is the probability of agreement by chance. If raters are in complete agreement then  $\kappa = 1$  and no agreement other than by chance will result in  $\kappa = 0$ . The value would be negative if there is no effective agreement among the raters or the agreement is worse than random. Additionally we also show accuracy (ACC) and per-class F1 scores. For all metrics, the higher the better.

### 3.4. Results

We study the effect of training ViPTT-Net for the task of tuberculosis type classification using different training strategies:

**No PT.** ViPTT-Net is trained from scratch on the 732 CT scans annotated for tuberculosis types

**PT.** ViPTT-Net is first pretrained on a subset of the UCF50 dataset which around 1300 video clips annotated for human activities. Then the last layer is replaced with a five unit softmax and fine-tuned on 732 CT scans annotated for tuberculosis types.

**Table 1**

Overall Kappa score and per-class F1 score achieved by different methods on the validation dataset of tuberculosis type classification. FibC denotes *Fibro-cavernous*. For all methods, the ViPTT-Net model is used.

Method	Kappa	Infiltrative	Focal Tuberculoma	Miliary	FibC
No PT	0.17	0.61	0.46	0.0	0.14
PT	<b>0.35</b>	<b>0.68</b>	<b>0.56</b>	0.09	0.48
PT+CW	0.30	0.59	0.41	<b>0.37</b>	<b>0.65</b>
PT+CW+AUG	0.33	0.59	0.47	0.27	<b>0.54</b>

**Table 2**

Overall Kappa and Accuracy score achieved by different methods on the test set by ImageCLEF 2021 Tuberculosis - TBT classification.

Method	Kappa	Accuracy
PT	0.13	42.30
PT+CW	0.14	38.50
PT+CW+AUG	<b>0.20</b>	<b>42.30</b>

**PT+CW.** Same as PT, but additionally weighted loss is used with weights computed over the training set of 732 CT scans annotated for tuberculosis types

**PT+CW+AUG.** Same as PT+CW, but additionally the 3D volumes are randomly rotated during training.

Table 1 summarizes the results which show that pretraining ViPTT-Net on a subset of the UCF50 dataset followed by fine-tuning for tuberculosis type classification (PT) improves kappa score by 0.18 on the validation set compared to training ViPTT-Net from scratch (No PT). In both configurations (No PT and PT), F1 score is the lowest for Tuberculoma class with scores of 0 and 0.09 for No PT and PT respectively. This is improved by using weighted loss (PT+CW), where the model achieves Tuberculoma F1 score 0.37. The F1 score of PT+CW also improves for FibC class by a large margin compared to No PT, although there is a slight drop in overall kappa score. PT+CW with data augmentation (PT+CW+AUG) further improves performance, more specifically for Miliary class by a large margin.

We also report results on the final test set by ImageCLEF 2021 - TBT classification task in Table 2. Using weighted loss results in similar performance improvements. Our best method PT+CW+AUG in which ViPTT-Net is trained to minimize the weighted cross entropy loss and data augmentation performs the best with a kappa score of 0.20 and accuracy of 42.30%.

## 4. Discussion and Conclusion

We address the problem of predicting tuberculosis types from 3D chest CT scans. We develop a hybrid CNN-RNN model, termed ViPTT-Net, which is capable of learning both spatial and temporal features of the CT scan. Our experiments demonstrate that pretraining on human action recognition task significantly improves performance of tuberculosis type classification rather

than training the model from scratch. This is most significant for Miliary and Fibro-cavernous types without specifically aiming to improve performance for those classes. Interestingly, these classes along with Tuberculoma are the tuberculosis types which have the least amount of samples in the dataset. To further deal with class imbalance, we use a weighted loss function with weights computed over the training set that improves performance of under-represented classes significantly. Data augmentation also improved performance for few tuberculosis types on the validation set. On the test set, the highest kappa score was observed when pretraining ViPTT-Net on videos, using weighted loss function and data augmentation. This method achieved 2nd place in the ImageCLEF 2021 Tuberculosis - TBT classification task which operates on the CT image alone without using the additional patient meta-data and the lung segmentation masks.

## References

- [1] P. Lakhani, B. Sundaram, Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks, *Radiology* 284 (2017) 574–582.
- [2] P. Rajpurkar, C. O’Connell, A. Schechter, N. Asnani, J. Li, A. Kiani, R. L. Ball, M. Mendelson, G. Maartens, D. J. van Hoving, et al., CheXaid: deep learning assistance for physician diagnosis of tuberculosis using chest x-rays in patients with hiv, *NPJ digital medicine* 3 (2020) 1–8.
- [3] M. Nash, R. Kadavigere, J. Andrade, C. A. Sukumar, K. Chawla, V. P. Shenoy, T. Pande, S. Huddart, M. Pai, K. Saravu, Deep learning, computer-aided radiography reading for tuberculosis: a diagnostic accuracy study from a tertiary hospital in india, *Scientific reports* 10 (2020) 1–10.
- [4] X. Li, Y. Zhou, P. Du, G. Lang, M. Xu, W. Wu, A deep learning system that generates quantitative CT reports for diagnosing pulmonary tuberculosis, *Applied Intelligence* (2020) 1–12.
- [5] H. Zunair, A. Rahman, N. Mohammed, J. P. Cohen, Uniformizing techniques to process CT scans with 3D CNNs for tuberculosis prediction, in: *International Workshop on Predictive Intelligence In Medicine*, Springer, 2020, pp. 156–168.
- [6] H. Zunair, A. Rahman, N. Mohammed, Estimating severity from CT scans of tuberculosis patients using 3D convolutional nets and slice selection., in: *CLEF (Working Notes)*, 2019.
- [7] V. Liauchuk, ImageCLEF 2019: Projection-based CT image analysis for TB severity scoring and CT report generation, *CLEF2019 Working Notes* 2380 (2019) 9–12.
- [8] X. W. Gao, R. Hui, Z. Tian, Classification of CT brain images based on deep learning networks, *Computer methods and programs in biomedicine* 138 (2017) 49–56.
- [9] M. Grewal, M. M. Srivastava, P. Kumar, S. Varadarajan, RADnet: Radiologist level accuracy using deep learning for hemorrhage detection in CT scans, in: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018, pp. 281–284.
- [10] A. Gentili, ImageCLEF2019: Tuberculosis-severity scoring and CT report with neural networks, transfer learning and ensembling, *CLEF2019 Working Notes* 2380 (2019) 9–12.
- [11] A. Hamadi, N. B. Cheikh, Y. Zouatine, S. M. B. Menad, M. R. Djebbara, ImageCLEF 2019:

- Deep learning for tuberculosis CT image analysis, CLEF2019 Working Notes 2380 (2019) 9–12.
- [12] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, et al., Gradient-based learning applied to document recognition, *Proceedings of the IEEE* 86 (1998) 2278–2324.
- [13] A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, in: *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.
- [14] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [15] S. Ebrahimi Kahou, V. Michalski, K. Konda, R. Memisevic, C. Pal, Recurrent neural networks for emotion recognition in video, in: *Proceedings of the 2015 ACM on international conference on multimodal interaction*, 2015, pp. 467–474.
- [16] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, S. Valaee, Recent advances in recurrent neural networks, *arXiv preprint arXiv:1801.01078* (2017).
- [17] H. Salehinejad, E. Ho, H.-M. Lin, P. Crivellaro, O. Samorodova, M. T. Arciniegas, Z. Merali, S. Suthiphosuwana, A. Bharatha, K. Yeom, et al., Deep sequential learning for cervical spine fracture detection on computed tomography imaging, *arXiv preprint arXiv:2010.13336* (2020).
- [18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in: *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.
- [19] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: *International Conference on Learning Representations*, 2015.
- [20] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, J. Schmidhuber, LSTM: A search space odyssey, *IEEE transactions on neural networks and learning systems* 28 (2016) 2222–2232.
- [21] K. K. Reddy, M. Shah, Recognizing 50 human action categories of web videos, *Machine vision and applications* 24 (2013) 971–981.
- [22] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning, *Proceedings of the IEEE* 109 (2020) 43–76.
- [23] H. Zunair, A. B. Hamza, Melanoma detection using adversarial training and deep transfer learning, *Physics in Medicine & Biology* 65 (2020) 135005.
- [24] B. Ionescu, H. Müller, R. Peteri, A. Ben Abacha, M. Sarrouti, D. Demner-Fushman, S. A. Hasan, S. Kozlovski, V. Liauchuk, Y. Dicente, V. Kovalev, O. Pelka, A. G. S. de Herrera, J. Jacutprakart, C. M. Friedrich, R. Berari, A. Tauteanu, D. Fichou, P. Brie, M. Dogariu, L. D. Ștefan, M. G. Constantin, J. Chamberlain, A. Campello, A. Clark, T. A. Oliver, H. Moustahfid, A. Popescu, J. Deshayes-Chossart, Overview of the ImageCLEF 2021: Multimedia retrieval in medical, nature, internet and social media applications, in: *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, *Proceedings of the 12th International Conference of the CLEF Association (CLEF 2021)*, LNCS Lecture Notes in Computer Science, Springer, Bucharest, Romania, 2021.
- [25] S. Kozlovski, V. Liauchuk, Y. Dicente Cid, V. Kovalev, H. Müller, Overview of ImageCLEFt-tuberculosis 2021 - CT-based tuberculosis type classification, in: *CLEF2021 Working Notes*,

CEUR Workshop Proceedings, CEUR-WS.org <<http://ceur-ws.org>>, Bucharest, Romania, 2021.

- [26] G. Van Rossum, et al., Python programming language., in: USENIX annual technical conference, volume 41, 2007, p. 36.
- [27] F. Chollet, et al., Keras, <https://keras.io>, 2015.
- [28] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: A system for large-scale machine learning, in: 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16), 2016, pp. 265–283.