Kandel and Castelli *Health Inf Sci Syst (2021) 9:33*
https://doi.org/10.1007/s13755-021-00163-7

Health Information Science and Systems

RESEARCH

Improving convolutional neural networks performance for image classification using test time augmentation: a case study using MURA dataset

Ibrahem Kandel* and Mauro Castelli

Abstract

Bone fractures are one of the main causes to visit the emergency room (ER); the primary method to detect bone fractures is using X-Ray images. X-Ray images require an experienced radiologist to classify them; however, an experienced radiologist is not always available in the ER. An accurate automatic X-Ray image classifier in the ER can help reduce error rates by providing an instant second opinion to the emergency doctor. Deep learning is an emerging trend in artificial intelligence, where an automatic classifier can be trained to classify musculoskeletal images. Image augmentations techniques have proven their usefulness in increasing the deep learning model's performance. Usually, in the image classification domain, the augmentation techniques are used during training the network and not during the testing phase. Test time augmentation (TTA) can increase the model prediction by providing, with a negligible computational cost, several transformations for the same image. In this paper, we investigated the effect of TTA on image classification performance on the MURA dataset. Nine different augmentation techniques were evaluated to determine their performance compared to predictions without TTA. Two ensemble techniques were assessed as well, the majority vote and the average vote. Based on our results, TTA increased classification performance significantly, especially for models with a low score.

Keywords: Image classification, Convolutional neural networks, Transfer learning, Test time augmentation, Deep learning, Ensemble learning

Introduction

Musculoskeletal X-ray images are crucial for fracture classification. Usually, when a patient has an accident or suspects a fracture, the patient goes to the emergency room (ER), where the ER doctor will first do an X-ray to detect fractures. The misclassification rate of X-ray images in ER is very high due to several factors, like the fact that the ER doctor classifying the X-ray is not an experienced radiologist and the rapidness of the process that leads to mistakes [1]. An automatic classifier to assist the doctor in classifying X-ray images can be a great help and can reduce the error rate [2]. Deep learning is a subfield of artificial intelligence composed mainly of artificial neural networks (ANN). Convolutional neural networks (CNN) are ANN with at least one convolution layer. Due to its robustness and its state-of-the-art (SOTA) results, it becomes the default classifier for the computer vision domain. To accurately train a CNN, usually, enormous image datasets are required. In the medical field, it is usually impossible to find a dataset with millions of images. Many methods were introduced in the literature to tackle this problem, like using transfer learning [3–6] or image augmentation techniques. Image augmentation uses several iterations from the same image to increase the dataset's size and train the model on different image transformations.

*Correspondence: D20181143@novaims.unl.pt
Nova Information Management School (NOVA IMS), Universidade Nova de Lisboa, Campus de Campolide, 1070-312 Lisboa, Portugal

Geometric transformations, among other techniques, were introduced. Usually, image augmentation is used for image classification during the training time but not during the prediction time (test time). Test time augmentation (TTA) refers to the usage of image augmentation techniques during prediction time to increase the models' robustness.

As pointed out by Shorten and Khoshgoftaar [7], image augmentation can help in unbalanced problems by increasing the number of observations in underrepresented classes [8–10]. Other authors used image augmentation during the training phase to increase classifier performance [11, 12]. Rane et al. [13] investigated the effect of ensemble learning on classifying histopathology images. They used TTA to improve the model robustness and they applied the same nine augmentation techniques for training and testing. The authors averaged the results of the TTA operations into one final score. However, they reported only the results using TTA but did not report the model's results without TTA, thus making it difficult to understand the potential of TTA. Wang et al. [14] used TTA to estimate the model's uncertainty for segmenting fetal brain images. They reported that TTA did improve the segmentation results as long as it can calculate the segmentation model uncertainty.

Amiri et al. [15] used TTA to improve the performance of breast image segmentation. In their work, they applied a shifting augmentation technique with values ranging from − 25 pixels to + 25 pixels. Experimental results showed that TTA provides a robust method to determine the stability of the detector. Sigurthorsdottir et al. [16] used TTA to increase CNN's and RNN's performance in classifying ECG signals. They used ten different augmentation techniques and then took the average of these results as a final score. They reported that TTA did improve the results of the model compared to the model without TTA. Wang et al. [17] used TTA to improve the segmentation of brain tumor images. They considered flipping, rotation, and scaling as augmentation techniques and they tested the effect of TTA on 3D UNet, WNet, and cascaded networks. In all the experiments, TTA did improve the results compared to the same models without TTA.

Typically, the TTA is used in image segmentation, and as far as we know, there are very few studies that thoroughly studied the effect of TTA specifically for image classification.

In this paper, we investigated the usage of TTA for increasing the performance of image classification. To do so, we applied nine different geometric techniques and assessed their performance. Also, we combined the prediction of these nine transformations by using average voting and majority voting techniques.

The rest of the paper is organized as follows: Sect. 2 presents the methodology and the dataset used. In Sect. 3, we present the results achieved. In Sect. 4, we present a discussion about the results obtained. In Sect. 5, we conclude the paper by summarizing the main findings of this work.

Methodology

In this section, we discuss the methods used in this paper.

Convolutional Neural Networks

Convolutional neural networks (CNNs) have become the de-facto algorithm for many computer vision tasks in recent years. One of the many advantages of CNNs is the concept of weight sharing, where instead of connecting all neurons (like in fully connected neural networks), a kernel can be used to map the features. Using weight sharing decreases the network size significantly and make it more robust against overfitting. The convolution operation is the operation that distinguishes CNNs from others neural networks. The convolution is a linear mathematical operation where a kernel (filter) is used to map the input by multiplying the inputs by a set of weights. The convolution operation result is a feature map that will be used instead of the input. The convolution operation is shown in Eq. (1):

$$O[i,j] = F(u,v) * I(i,j) = \sum_u \sum_v \sum_{c\in\{R,G,B\}} F_c(u,v) \odot I_c(i+u, j+v) \tag{1}$$

where $I(.)$ is the input image, c is the color channels, $F(u,v)$ is the kernel, and $O(i,j)$ is the output feature map in the (i,j) position.

There have been many architectures that were introduced in the literature. In this paper, we will use the following SOTA CNN: VGG19, InceptionV3, ResNet50, Xception, and DenseNet121. All the CNNs used were pre-trained on the ImageNet dataset.

VGG19

A group of researchers introduced VGG CNN [18] from Oxford university to participate in the ImageNet challenge in 2014, and it achieved second place. VGG consists of several convolution blocks separated by a dropout layer. Each convolution block consists of three or four convolutional layers sequentially connected. Several versions were introduced by the authors of the VGG, which varies in the number of convolution layers. We will use the VGG19 version.

InceptionV3

InceptionV3 CNN [19] was introduced by a group of researchers from Google to participate in the ImageNet Challenge in the same year as VGG. InceptionV3 achieved a higher result than VGG and achieved first place. One of the main differences between them is the inception module's presence to decrease the computational power needed and capture different aspect ratios from the same image. There are several versions of the Inception networks. In this paper, we will use the InceptionV3 version.

ResNet50

ResNet CNN [20] was introduced by a group of researchers from Microsoft to participate in the ImageNet challenge in 2015, and it achieved first place in that year. One of the main perks of ResNet that separated it from other networks is the presence of residual connections. The authors have noticed that the performance deteriorates rapidly by increasing the CNN's depth, mainly because of the vanishing gradient problem. The authors proposed a connection that will act as a shortcut connection that will escape several layers each time. There are several versions of the ResNet networks. In this paper, we will use the ResNet50 version.

Xception

Xception CNN [21] was introduced by Francois Chollet in 2017. The Xception network was inspired by both the Inception module and the residual connection. The author replaced the conventional convolution layer with a depthwise convolution layer, which significantly decreased the computational power needed to train the network. The performance obtained by the Xception network for the ImageNet dataset is better than both the VGG19 and InceptionV3 and comparable to ResNet50.

DenseNet121

DenseNet CNN [22] was introduced in 2017. The residual connection from ResNet CNN inspired the DenseNet network. In DenseNet CNN, to overcome the vanishing gradient problem and reuse the subsequent convolution layers' features, the authors densely connected the subsequent convolution layers. The authors concatenated the results instead of adding them like the authors of ResNet. The result of DenseNet for the ImageNet dataset is higher than all the CNN mentioned above.

Test time augmentation

Test time augmentation (TTA) refers to the use of many variants of images during test time to provide different predictions for the same image [7, 23]. The results of the TTA can be combined in various ways, like taking the average vote or taking the majority vote of all the variants. Nine different augmentation techniques were studied in this paper, namely, horizontal flip; vertical flip; 40% zooming; 180° rotation; horizontal flip with vertical flip (H_V); horizontal flip with rotation (H_R); vertical flip with rotation (V_R); horizontal flip, vertical flip, and rotation (H_V_R); and combining all four methods, horizontal flip, vertical flip, rotation, and zooming (H_V_R_Z). We studied each technique's results alone and two combination methods: average votes and majority votes. The average vote considers the average of the scores obtained by a CNN network after the augmentation techniques (i.e., in our case, the average of nine values), and it outputs the predicted label based on this value. On the other hand, in the case of the majority vote, the predictions (obtained by each augmentation technique) for each label are summed, and the label with the majority vote is predicted. Thus, the former combination strategy considers the scores of the CNNs, while the latter technique works by directly considering the predicted labels.

Dataset and the evaluation metric

The dataset used in this paper is the MURA dataset [24], a publicly available dataset composed of X-Ray images of seven different upper extremities organs, namely, finger, wrist, hand, forearm, elbow, humerus, and shoulder. The size of the images is different and ranges from 117×512 pixels to 512×512 pixels. The authors of the dataset divided the images into two partitions: the training dataset and the testing dataset. The training dataset has 36,808 images, and the testing dataset has a total of 3,197 images. The MURA dataset is considered particularly challenging because of the inconsistency of the image's sizes, the presence of unbalanced classes in some organ datasets, and the small size of other organ datasets. A summary statistic about the MURA dataset is shown in Table 1. The evaluation metric proposed by the authors of the dataset is the Kappa metric [25]. The Kappa metric

Table 1 MURA dataset summary

Category	Training dataset		Test dataset	
	Normal	Fractured	Normal	Fractured
Wrist	5765	3987	364	295
Shoulder	4211	4168	285	278
Hand	4059	1484	271	189
Finger	3138	1968	214	247
Elbow	2925	2006	235	230
Forearm	1164	661	150	151
Humerus	673	599	148	140
Total	21,935	14,873	1667	1530

is a prevalent metric specially used for imbalanced classification problems. The Kappa metric ranges from [− 1,1] where −1 means a completely random classifier and +1 means a perfect classifier. Kappa metric will be used to evaluate the results obtained to be consistent with other studies like [24, 26, 27]. The MURA dataset is available from http://arxiv.org/abs/1712.06957 (Fig. 1).

Results

The training dataset was split to 80%/20% for training and validation, respectively. Throughout all the experiments, all the hyperparameters were fixed. All the CNN used were pre-trained on the ImageNet dataset, i.e., transfer learning was used. The batch size used was 64, and all the images were resized to 96 × 96 pixels. Adam optimizer [28] was used with a learning rate of 0.0001. Because the MURA dataset is a binary classification task, binary cross-entropy was used as the loss function. Early stopping of 50 epochs was used to halt the training if no performance increase happens to the validation dataset.

Four augmentation methods were considered during the training phase, namely, horizontal flip, vertical flip, rotation, and zooming. The hyperparameters used during the training phase are shown in Table 2. During test time, the nine augmentation techniques stated early were applied. In this study, we investigated the performance of each of these nine techniques and their average vote and the majority vote. To mitigate the effect of the algorithm's stochastic nature and to produce the confidence interval, the Kappa mean score and the confidence interval were created by repeating each experiment 50 times. A schematic diagram of the performed experiments is shown in Fig. 2.

Table 2 The hyperparameters were used for all the experiments

Optimizer	Adam
Learning rate	0.0001
Loss function	Binary Cross-entropy
Early stopping	50 epochs
Batch size	64
Validation split	20%

Finger images

We applied nine different geometric image augmentation techniques for the finger images and taken the average vote and the majority vote of the different methods. The results are presented in Table 3. For the VGG19 network, the original model without any TTA yielded a Kappa score of 0.3944. The Kappa score of horizontal, vertical, and H_V augmentation techniques was lower than the original model. The Kappa score of the remaining augmentation techniques was higher than the original model. The rotation augmentation technique achieved the highest score among the nine different augmentation techniques with a Kappa score of 0.4333. The average score and the majority vote score of the nine different techniques were higher than the original score.

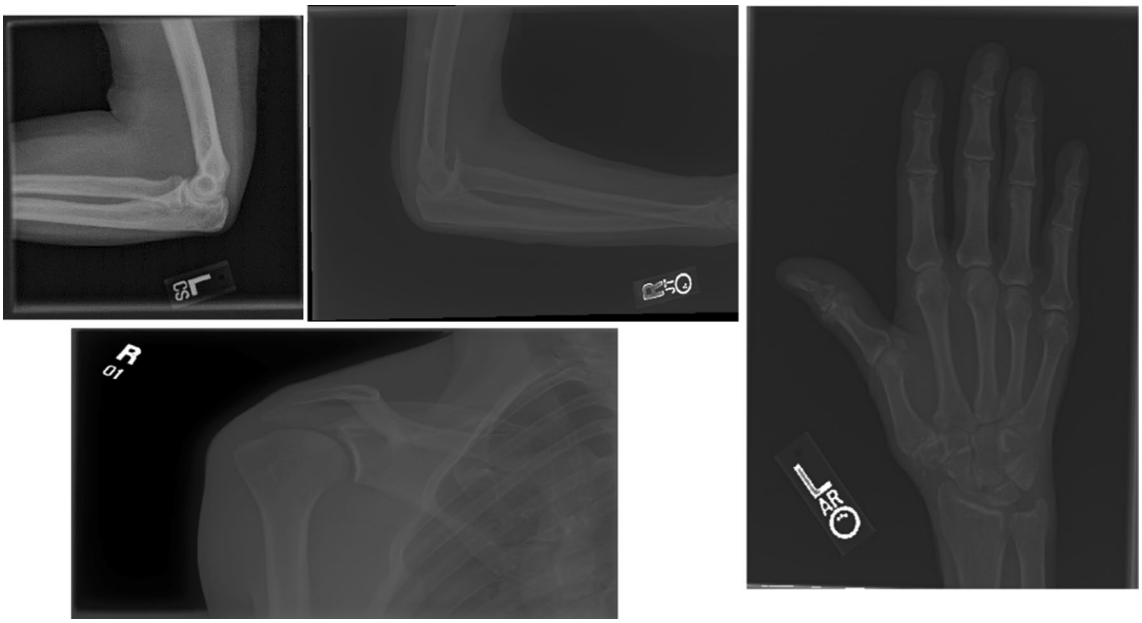

Fig. 1 This figure shows a sample of the MURA dataset

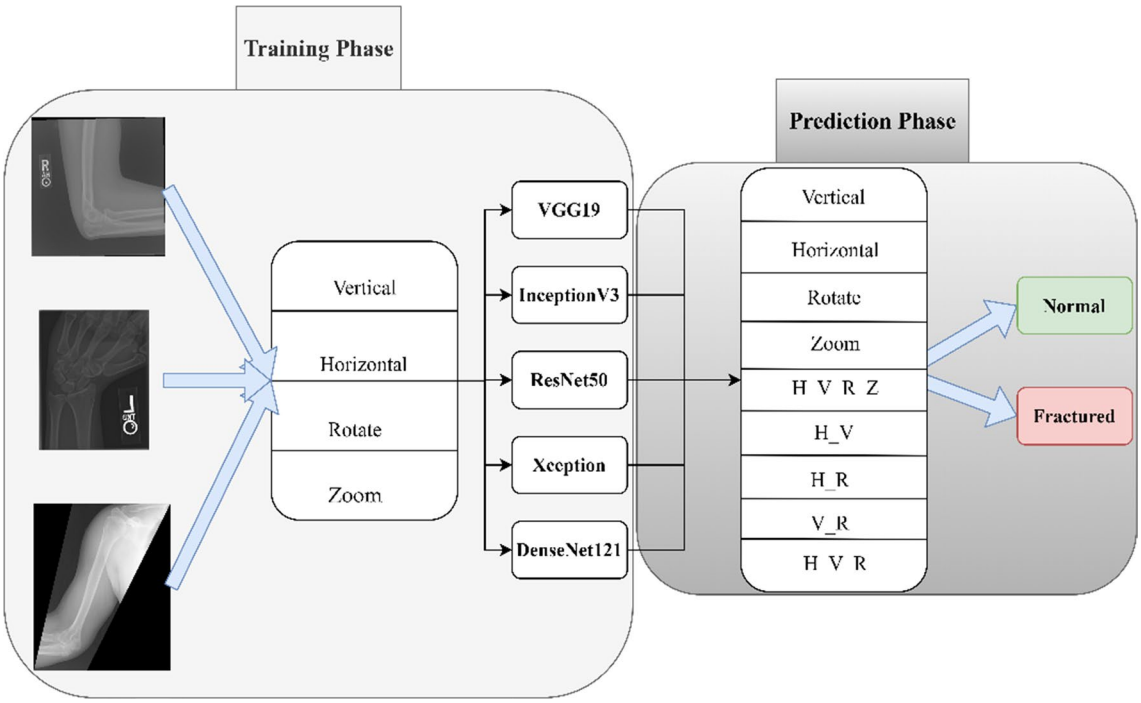

Fig. 2 The experiment executed in this study

Table 3 The average Kappa score of test time augmentation of Finger images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	0.3944	0.3550	0.3705	0.3891	0.3840
Horizontal	0.3883 ± 0.32%	0.3475 ± 0.50%	0.3623 ± 0.49%	0.4009 ± 0.43%	0.3751 ± 0.39%
Vertical	0.3813 ± 0.37%	0.3388 ± 0.46%	0.3702 ± 0.20%	0.4086 ± 0.41%	0.4110 ± 0.51%
Rotate	0.4333 ± 0.48%	0.4273 ± 0.59%	0.4402 ± 0.63%	0.4883 ± 0.61%	0.4586 ± 0.57%
Zoom	0.4220 ± 0.67%	0.4646 ± 0.67%	0.4458 ± 0.68%	0.4497 ± 0.71%	0.4483 ± 0.72%
H_V_R_Z	0.4000 ± 0.65%	0.4246 ± 0.75%	0.4328 ± 0.64%	0.4607 ± 0.67%	0.4487 ± 0.79%
H_V	0.3765 ± 0.45%	0.3454 ± 0.67%	0.3572 ± 0.69%	0.4161 ± 0.64%	0.3875 ± 0.54%
H_R	0.4273 ± 0.65%	0.4371 ± 0.67%	0.4423 ± 0.76%	0.4774 ± 0.64%	0.4646 ± 0.57%
V_R	0.4313 ± 0.65%	0.4346 ± 0.67%	0.4522 ± 0.76%	0.4811 ± 0.62%	0.4505 ± 0.58%
H_V_R	0.4328 ± 0.62%	0.4251 ± 0.70%	0.4470 ± 0.75%	0.4748 ± 0.73%	0.4616 ± 0.61%
Average vote	0.4291 ± 0.37%	0.4546 ± 0.41%	0.4624 ± 0.29%	0.5055 ± 0.36%	0.4712 ± 0.31%
Majority vote	0.4240 ± 0.39%	0.4525 ± 0.38%	0.4570 ± 0.35%	0.4955 ± 0.39%	0.4682 ± 0.39%

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Overall, the rotation augmentation technique produced an increase of 9.87% over the normal method, followed by the average vote with an increase of 8.82% compared to the original method without any TTA. For the InceptionV3 network, the normal method, without any TTA, achieved a Kappa score of 0.3550.

The score of horizontal, vertical, and H_V augmentation techniques was less than the original method. The rest of the techniques scored higher than the original method. The zoom technique achieved the highest score among the nine different augmentation techniques with a Kappa score of 0.4646. The average vote score and the majority vote were higher than both the original and the different techniques except for the zoom technique.

Overall, for the InceptionV3 network, the highest score achieved was obtained with the zoom technique, with a

Kappa score of 0.4646, representing a 30.87% increase over the original score. The average vote achieved an increase of 28.06%, and the majority vote achieved an increase of 27.47%. For the ResNet50 network, the original method, without any TTA, yielded a Kappa score of 0.3705. The Kappa score of the Horizontal and H_V augmentation techniques was lower than the original method. The vertical method score was slightly lower than the original method. The rest of the augmentation techniques achieved a higher Kappa score than the original method. The highest Kappa score, among the nine different augmentation techniques, was achieved by the V_R technique. The average vote score and the majority vote of the augmentation techniques were higher than the original method (without TTA) and the nine different augmentation techniques.

Overall, the average vote highest score was obtained with a Kappa score of 0.4624, which represents an increase of 24.81% to the original method. For the Xception network, the original model score was 0.3891. All nine different augmentation techniques yielded better results than by using the original method. The rotation technique achieved the highest score among the different methods with a Kappa score of 0.4883, representing a 25.51% increase over the original method. The average vote and the majority vote scores were higher than the original method and the nine different augmentation techniques. Overall, for the Xception network, the average vote achieved the best score with a Kappa score of 0.5055, which represents an increase of 29.94% over the original method.

For the DenseNet121 network, the original method scored a Kappa score of 0.3840. Only the horizontal augmentation technique score was lower than it. All the other augmentation techniques achieved better results than the original model. The H_R augmentation technique achieved the highest score with a Kappa score of 0.4646, representing a 20.98% increase over the original method. The average vote and the majority vote achieved better results than the original method and the nine different augmentation techniques alone. Overall, the best score achieved for the DenseNet121 was 0.4712, which was achieved by the average vote. This value represents a 22.71% increase over the original method.

By comparing only the average vote and the majority vote scores to the original method, the best performance was achieved by the Xception network with a difference of 29.94% for the average vote and 27.36% for the majority vote. The lowest performance was obtained by the VGG19 network, with a difference of 8.82% for the average vote and 7.51% for the majority vote. The average performance gain produced by the majority vote, considering all the networks, was 21.53%. On the other hand, considering all the networks, the average vote produced, on average, a 22.87% performance improvement. Figure 3 shows Kappa scores distributions of the 50 experiments of each network.

Humerus images

The results of the nine augmentation techniques, the majority vote, and the average vote for the humerus images are presented in Table 4. For the VGG19 network, the original method had a Kappa score of 0.6387. The H_V_R_Z augmentation technique achieved poorer performance than the original method. The remaining augmentation techniques outperformed the original method. The horizontal augmentation achieved the highest score among the augmentation techniques with a Kappa score of 0.6604. Both the average vote and the majority vote outperformed the original method and the augmentation techniques. Overall, for the VGG19 network, the majority vote achieved the highest Kappa score with a value of 0.6835, representing an increase of 7.01% over the original model.

It is worth noting that the average vote score was slightly lower than the majority vote with a Kappa score of 0.6792, which represents an increase of 6.34% over the original method. Concerning the InceptionV3 network, the vertical, zoom, and H_V augmentation techniques achieved a higher score than the original method, while the Kappa score of the remaining augmentation techniques was lower than the original method. The zoom augmentation technique produced the highest Kappa score among the considered augmentation techniques with a value of 0.6282. The average vote and the majority vote scores were higher than both the original method and the nine different augmentation methods. Overall, the best performance was achieved with the average vote, with a Kappa score of 0.6677, representing an increase of 9.21% over the original method.

For the ResNet50 network, the Kappa score of the original method was 0.5784. All the augmentation techniques outperformed the original method; however, the horizontal method score was lower than the original method. The best Kappa score among the nine augmentation techniques was achieved by vertical flipping with a value of 0.6222. The average vote and the majority score were higher than the original method and the augmentation techniques. The best score was achieved by taking the average vote with a Kappa score of 0.6464, representing an increase of 11.76% over the original method.

For the Xception network, the original method had a Kappa score of 0.5964. The only method that scored lower than the original method was the vertical method. The H_R achieved the best score with a Kappa score of 0.6177. Both the average vote and the majority score

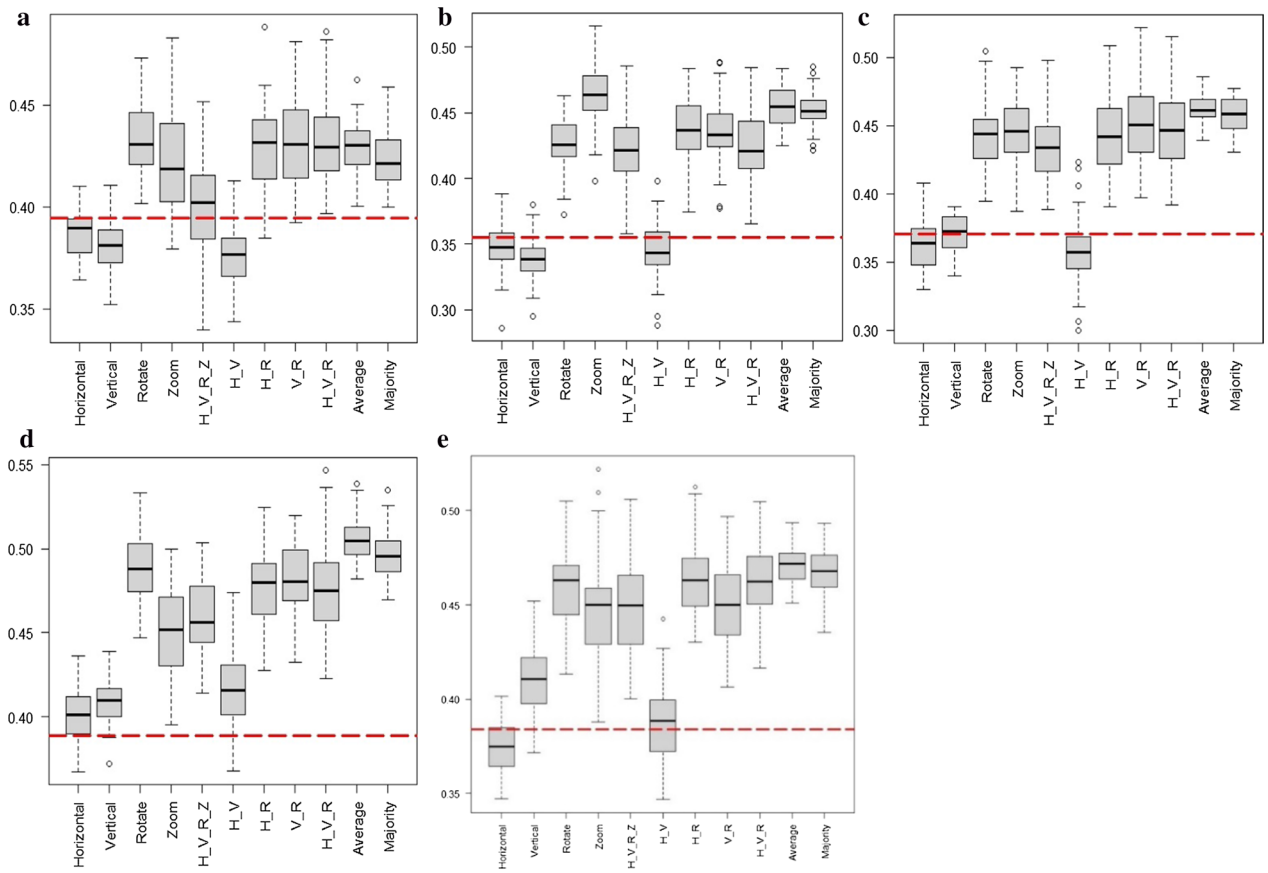

Fig. 3 This figure shows the Kappa scores distributions of the 50 experiments of each network for Finger images. **a** Kappa scores of VGG19 network; **b** Kappa scores of InceptionV3 network; **c** Kappa scores of ResNet50 network; **d** Kappa scores of Xception network; **e** Kappa scores of DenseNet121 network. The red line represents the score of each network without TTA

Table 4 Kappa score of test time augmentation of Humerus images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	0.6387	0.6114	0.5784	0.5964	0.5686
Horizontal	0.6604 ± 0.31%	0.6095 ± 0.58%	0.5725 ± 0.67%	0.6118 ± 0.48%	0.5459 ± 0.71%
Vertical	0.6449 ± 0.41%	0.6182 ± 0.67%	0.6222 ± 0.69%	0.5784 ± 0.56%	0.5967 ± 0.58%
Rotate	0.6405 ± 0.59%	0.6040 ± 0.72%	0.5975 ± 0.93%	0.6126 ± 0.82%	0.6109 ± 0.69%
Zoom	0.6317 ± 0.75%	0.6282 ± 0.82%	0.5924 ± 0.62%	0.5970 ± 0.76%	0.5826 ± 0.75%
H_V_R_Z	0.6295 ± 0.65%	0.6090 ± 1.02%	0.5819 ± 0.81%	0.6121 ± 0.73%	0.5985 ± 0.94%
H_V	0.6552 ± 0.49%	0.6153 ± 0.69%	0.5980 ± 0.63%	0.5897 ± 0.65%	0.5897 ± 0.81%
H_R	0.6477 ± 0.63%	0.6058 ± 0.94%	0.5990 ± 0.84%	0.6177 ± 0.69%	0.6211 ± 0.69%
V_R	0.6473 ± 0.56%	0.6042 ± 0.83%	0.5935 ± 0.80%	0.6120 ± 0.68%	0.6213 ± 0.81%
H_V_R	0.6377 ± 0.58%	0.6058 ± 0.55%	0.5954 ± 0.81%	0.6090 ± 0.84%	0.6189 ± 0.81%
Average vote	0.6792 ± 0.33%	0.6677 ± 0.50%	0.6464 ± 0.48%	0.6424 ± 0.39%	0.6466 ± 0.44%
Majority vote	0.6835 ± 0.31%	0.6612 ± 0.55%	0.6449 ± 0.41%	0.6411 ± 0.47%	0.6460 ± 0.58%

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

were higher than the original method. Overall, the highest Kappa score was achieved by taking the average vote, with a Kappa score of 0.6424, representing an increase of 7.72% over the original method. For the Dense121 network, the Kappa score of the original method was 0.5686. The horizontal method was the only method with a lower score than the original method.

The highest score among the different augmentation techniques was achieved by the V_R method, with a Kappa score of 0.6213. Both the average vote and the majority vote scores were higher than all the rest. Overall, the highest score for the DenseNet121 network was achieved by taking the average vote with a Kappa score of 0.6466, representing an increase of 13.72% compared to the original method. By comparing only the average vote and the majority vote scores to the original method, the most significant performance improvement was achieved on the DenseNet121 network with a difference of 13.72% for the average vote and 13.62% for the majority vote. The lowest performance gain was achieved on the VGG19 network, with a difference of 6.34% for the average vote and 7.01% for the majority vote. The majority vote method's average performance for all the networks was 9.56% and 9.75% for the average vote. Figure 4 shows Kappa scores distributions of the 50 experiments performed.

Forearm images

We performed nine different geometric image augmentation techniques for the forearm images and taken the average vote and the majority vote of the different methods. The results are presented in Table 5. The first one is the horizontal augmentation of the images; its score was slightly better than the original method for the VGG19 network and the Xception network and was better than the original method for the InceptionV3 network and better than the ResNet50 network by about 5%. The only network in which the score of TTA was lower than the original method was the DenseNet121 network. The first network is the VGG19; the Kappa score achieved without any TTA was 0.5552, and the performance of the

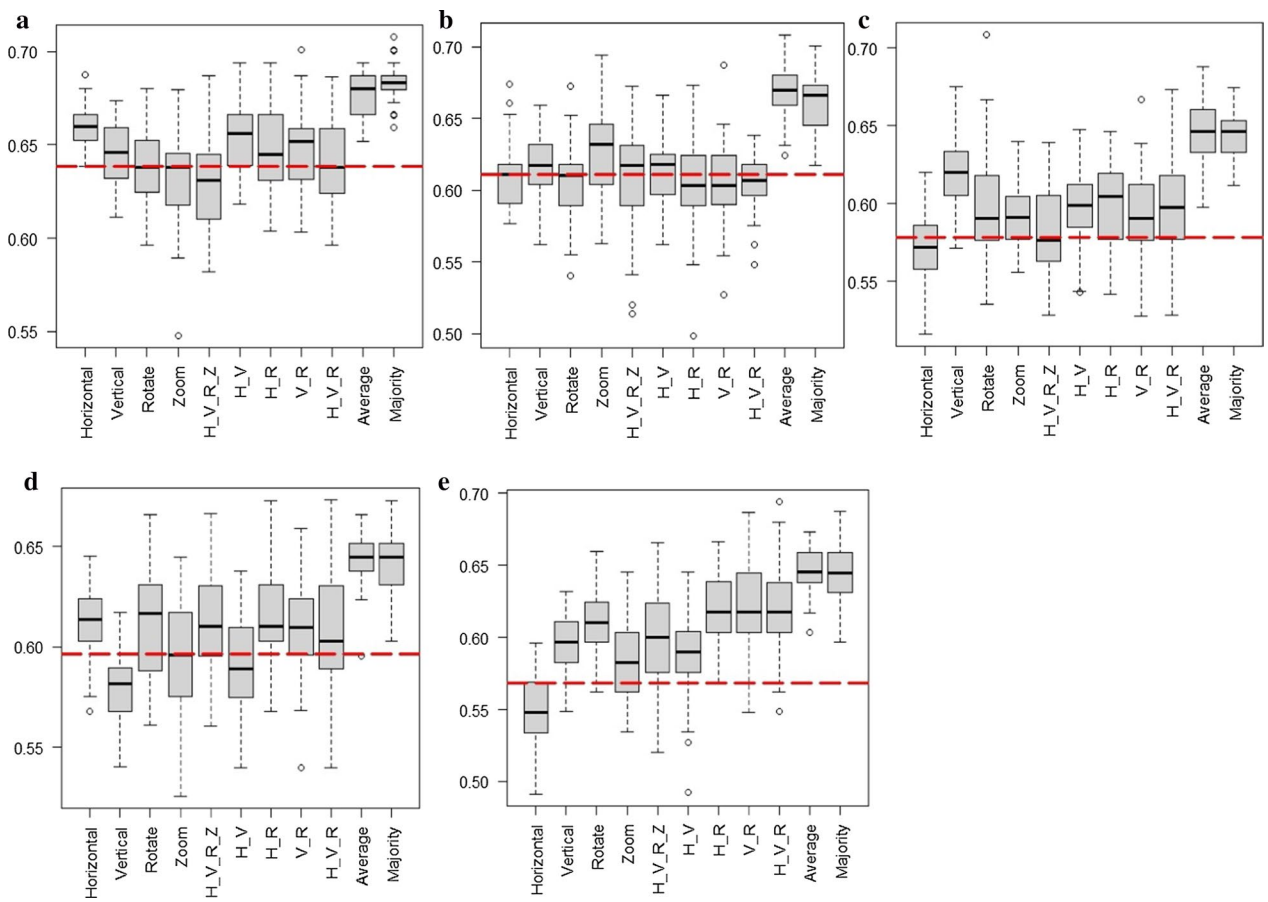

Fig. 4 This figure shows the Kappa scores distributions of the 50 experiments of each network for Humerus images. **a** Kappa scores of VGG19 network; **b** Kappa scores of InceptionV3 network; **c** Kappa scores of ResNet50 network; **d** Kappa scores of Xception network; **e** Kappa scores of DenseNet121 network. The red line represents the score of each network without TTA

Table 5 Kappa score of test time augmentation of Forearm images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	0.5552	0.5219	0.5750	0.4956	0.5420
Horizontal	0.5580 ± 0.49%	0.5502 ± 0.58%	0.6039 ± 0.53%	0.5021 ± 0.51%	0.5396 ± 0.5458%
Vertical	0.5465 ± 0.43%	0.5430 ± 0.65%	0.5792 ± 0.64%	0.4879 ± 0.38%	0.5459 ± 0.53%
Rotate	0.5493 ± 0.48%	0.5370 ± 0.85%	0.5154 ± 0.86%	0.4811 ± 0.65%	0.5187 ± 0.75%
Zoom	0.5122 ± 0.89%	0.5092 ± 0.86%	0.5293 ± 0.78%	0.4770 ± 0.84%	0.4837 ± 0.85%
H_V_R_Z	0.5214 ± 0.77%	0.5102 ± 0.99%	0.4955 ± 0.91%	0.4824 ± 0.96%	0.4802 ± 0.84%
H_V	0.5542 ± 0.53%	0.5664 ± 0.76%	0.5877 ± 0.63%	0.4895 ± 0.54%	0.5327 ± 0.63%
H_R	0.5461 ± 0.70%	0.5414 ± 0.82%	0.5284 ± 0.63%	0.4933 ± 0.74%	0.5235 ± 0.69%
V_R	0.541 ± 0.59%	0.5329 ± 0.86%	0.5150 ± 0.62%	0.4945 ± 0.70%	0.5161 ± 0.91%
H_V_R	0.5387 ± 0.54%	0.5391 ± 0.87%	0.5253 ± 0.75%	0.4911 ± 0.74%	0.5216 ± 0.78%
Average vote	0.5536 ± 0.34%	0.5760 ± 0.43%	0.5844 ± 0.42%	0.5143 ± 0.36%	0.5520 ± 0.40%
Majority vote	0.5599 ± 0.33%	0.5713 ± 0.56%	0.5899 ± 0.48%	0.5111 ± 0.46%	0.5526 ± 0.43%

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

horizontal augmentation was slightly better than using the method without any augmentation. The nine methods' average vote was worse than the original score; however, the majority vote score outperformed the original score. Overall, for VGG19, the TTA gave a minimal performance increase.

For the InceptionV3 network, only the kappa score of the zoom and the H_V_R_Z augmentations were lower than the original method. The remaining augmentations techniques did yield better performance than the original method. The highest score was achieved by the H_V augmentation technique. Overall, the highest Kappa score was achieved by taking the average of the nine augmentation techniques. For the ResNet50 network, the horizontal, vertical, and H_V methods yielded better results than the original method. However, the rest of the nine methods did not achieve better results than the original method. The average vote and the majority vote of the augmentation techniques achieved better results than the original method. Overall, for the ResNet50 network, the best Kappa score was achieved by the Horizontal augmentation method.

For the Xception network, the only augmentation technique that achieved better results than using the original method was the Horizontal technique with a Kappa score of 0.5021. The average vote and the majority vote of the nine different augmentation techniques yielded better results than using the model without TTA. Overall, for the Xception network, the best result was achieved by taking the average vote of all the different augmentation techniques.

For the DenseNet121 network, the only augmentation technique that yielded better results than using the original model was the Vertical flipping with a Kappa score of 0.5459 compared to the Kappa score of 0.5420 of the original models. The average vote and the majority vote did yield better results than by using the original model. Overall, for the DenseNet121 network, the best result was achieved by taking the majority vote.

Overall, by comparing only the average vote and the majority vote scores to the original method, the best performance was achieved by the InceptionV3 network with a difference of 10.38% for the average vote and 9.46% for the majority vote. The lowest performance was obtained by the VGG19 network with a difference of -0.28% for the average vote, and a difference of 0.84% for the majority vote. The majority vote method's average performance for all the networks was 3.60% and 3.48% for the average vote. Figure 5 shows Kappa scores distributions of the 50 experiments of each network.

Wrist images

The results of the augmentation techniques for the five networks for the wrist images are presented in Table 6. For the VGG19 network, the original model had a Kappa score of 0.5749. The vertical and H_V techniques achieved a lower score than the original method. The rest of the augmentation techniques achieved higher scores than the original method. The H_V_R technique achieved the highest score among the nine different techniques with a Kappa score of 0.6205. Both the average vote and the majority vote scores were higher than the original model. The highest score was achieved by taking the average vote with a Kappa score of 0.6359, representing an increase of 10.61% over the original method.

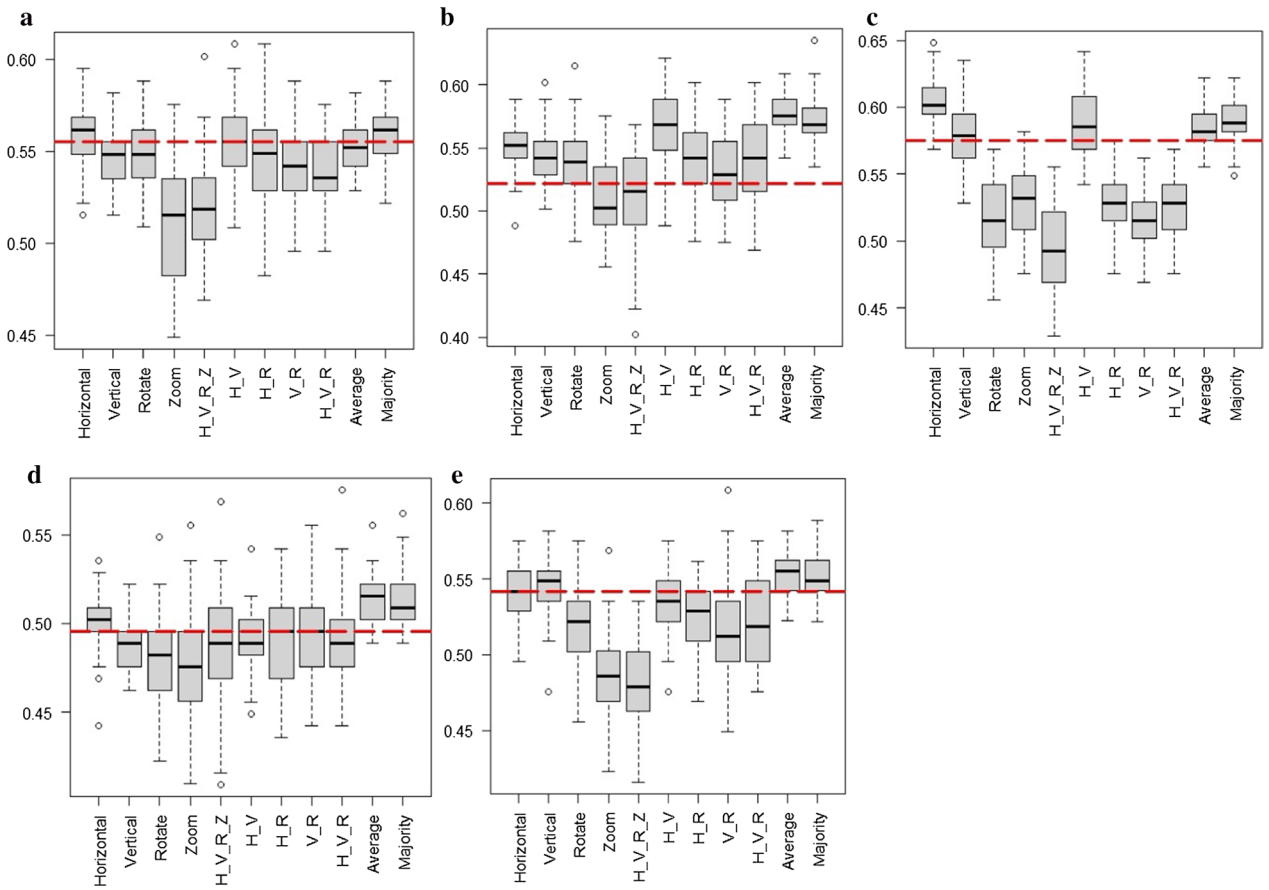

Fig. 5 This figure shows the Kappa scores distributions of the 50 experiments of each network for Forearm images. **a** Kappa scores of VGG19 network; **b** Kappa scores of InceptionV3 network; **c** Kappa scores of ResNet50 network; **d** Kappa scores of Xception network; **e** Kappa scores of DenseNet121 network. The red line represents the score of each network without TTA

Table 6 Kappa score of test time augmentation of Wrist images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	0.5749	0.6064	0.5551	0.6235	0.5250
Horizontal	0.5780 ± 0.31%	0.6101 ± 0.36%	0.5771 ± 0.34%	0.6246 ± 0.34%	0.5182 ± 0.29%
Vertical	0.5685 ± 0.30%	0.6138 ± 0.33%	0.5669 ± 0.32%	0.6182 ± 0.36%	0.5630 ± 0.32%
Rotate	0.6188 ± 0.45%	0.6174 ± 0.47%	0.6158 ± 0.41%	0.6085 ± 0.48%	0.6130 ± 0.44%
Zoom	0.6024 ± 0.45%	0.5929 ± 0.44%	0.5982 ± 0.44%	0.5946 ± 0.48%	0.5724 ± 0.49%
H_V_R_Z	0.5983 ± 0.39%	0.6063 ± 0.43%	0.5973 ± 0.50%	0.5998 ± 0.61%	0.5961 ± 0.60%
H_V	0.5700 ± 0.43%	0.6115 ± 0.46%	0.5873 ± 0.42%	0.6269 ± 0.38%	0.5496 ± 0.51%
H_R	0.6175 ± 0.40%	0.6134 ± 0.49%	0.6078 ± 0.43%	0.6074 ± 0.43%	0.6090 ± 0.56%
V_R	0.6194 ± 0.47%	0.6146 ± 0.49%	0.6068 ± 0.43%	0.6084 ± 0.42%	0.6116 ± 0.47%
H_V_R	0.6205 ± 0.41%	0.6158 ± 0.41%	0.6101 ± 0.46%	0.6114 ± 0.46%	0.6153 ± 0.54%
Average vote	0.6359 ± 0.24%	0.6487 ± 0.27%	0.6324 ± 0.27%	0.6382 ± 0.25%	0.6223 ± 0.27%
Majority vote	0.6334 ± 0.33%	0.6453 ± 0.29%	0.6292 ± 0.23%	0.6394 ± 0.25%	0.6159 ± 0.31%

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

For the InceptionV3 network, the original score was 0.6064. All the techniques score higher than the original method except for the zoom augmentation technique. The highest Kappa score among the nine augmentation techniques was achieved by the rotate augmentation technique with a value of 0.6174. The score of the average vote and the majority vote was higher than the original method. For the InceptionV3, the highest score was achieved by average vote with a Kappa score of 0.6487, representing an increase of 6.97% over the original method.

For the ResNet50 network, the original method had a Kappa score of 0.5551. All the augmentation techniques outperformed the original model. The rotate augmentation technique achieved the highest score among the augmentation techniques, with a Kappa score of 0.6158. The average vote and the majority vote scores were higher than the original method. Overall, For the ResNet50, the best score was achieved by taking the average vote with a Kappa score of 0.6324, representing 13.93% over the original method.

For the Xception network, the original model had a Kappa score of 0.6235. Only the horizontal and H_V augmentation methods scored higher than the original method. Overall, the H_V technique achieved the highest score among the nine augmentation methods, with a Kappa score of 0.6269. The average vote and the majority vote scores were higher than the original method. Overall, the majority vote achieved the highest Kappa score with a value of 0.6394, representing an increase of 2.56% over the original score.

For the DenseNet121, the original model had a Kappa score of 0.5250. The horizontal augmentation technique was the only technique with a lower score than the original model. The H_V_R achieved the highest Kappa score among the nine augmentation techniques with a Kappa score of 0.6153. Overall, by comparing only the average vote and the majority vote scores to the original method, the best performance was achieved by the DenseNet121 network with a difference of 18.54% for the average vote and 17.31% for the majority vote. The lowest performance gain was obtained by the Xception network, with a difference of 2.36% for the average vote and a difference of 2.56% for the majority vote. The majority vote method's average performance for all the networks was 9.96% and 10.48% for the average vote. Figure 6 shows Kappa scores distributions of the 50 experiments of each network.

Elbow images

The results of the nine different augmentation techniques with the average vote and the majority vote for the elbow images are presented in Table 7. For the VGG19 network, the Kappa score of the model without any augmentation was 0.6078. The zooming technique scored lower than the original method. All the other techniques yielded better performance than the original method. The highest score was achieved by the rotation technique, with a Kappa score of 0.6235. The average vote and the majority vote yield better results than the original model and the augmentation techniques alone. Overall, the best score was achieved by the average vote of the different augmentation techniques, with an increase of 4.14% over the original method.

For the InceptionV3 network, the normal method's Kappa score without any augmentation yielded a score of 0.6252. Rotation, H_R, V_R, and H_V_R augmentation techniques yielded better results than the original method; however, the Kappa score of the horizontal, zooming, H_V_R_Z, and H_V augmentation techniques was lower than the original method. The score of the average vote and the majority vote were higher than the original method. Overall, the best score was achieved by taking the average of all the nine augmentation techniques that yielded a Kappa score of 0.6810, which was an increase of 8.94% over the original method.

For the ResNet50 network, the Kappa score of the original method is 0.5908. The Kappa score of the horizontal, vertical, and H_V augmentation methods was lower than the original method. The rest of the nine augmentation techniques did score better than the original method, with the highest being the rotation technique that yielded a kappa score of 0.6317. The average vote and the majority vote scores were higher than the original model. Overall, for the ResNet50 network, the best score was achieved by taking the average vote of the nine different augmentation techniques, which yielded an increase of 11.03% over the original method.

For the Xception network, the Kappa score of the original method was 0.6336. The zooming augmentation technique was the only method that scored lower than the original method. All the remaining augmentation techniques' scores were higher than the original method. The H_V_R method was the best augmentation method, with a Kappa score of 0.6668. Both the average vote and the majority vote achieved better than the rest of the augmentation techniques. The majority vote achieved the best score, with a Kappa score of 0.6947, with an increase of 9.65% over the original method. It worth noting that the score of the average vote was slightly lower than the majority vote.

For the DenseNet121 network, the original model achieved a Kappa score of 0.6123. All the augmentation techniques achieved better scores than the original method, with only the horizontal technique score that is lower than the original method. The V_R augmentation technique score was the highest among the nine

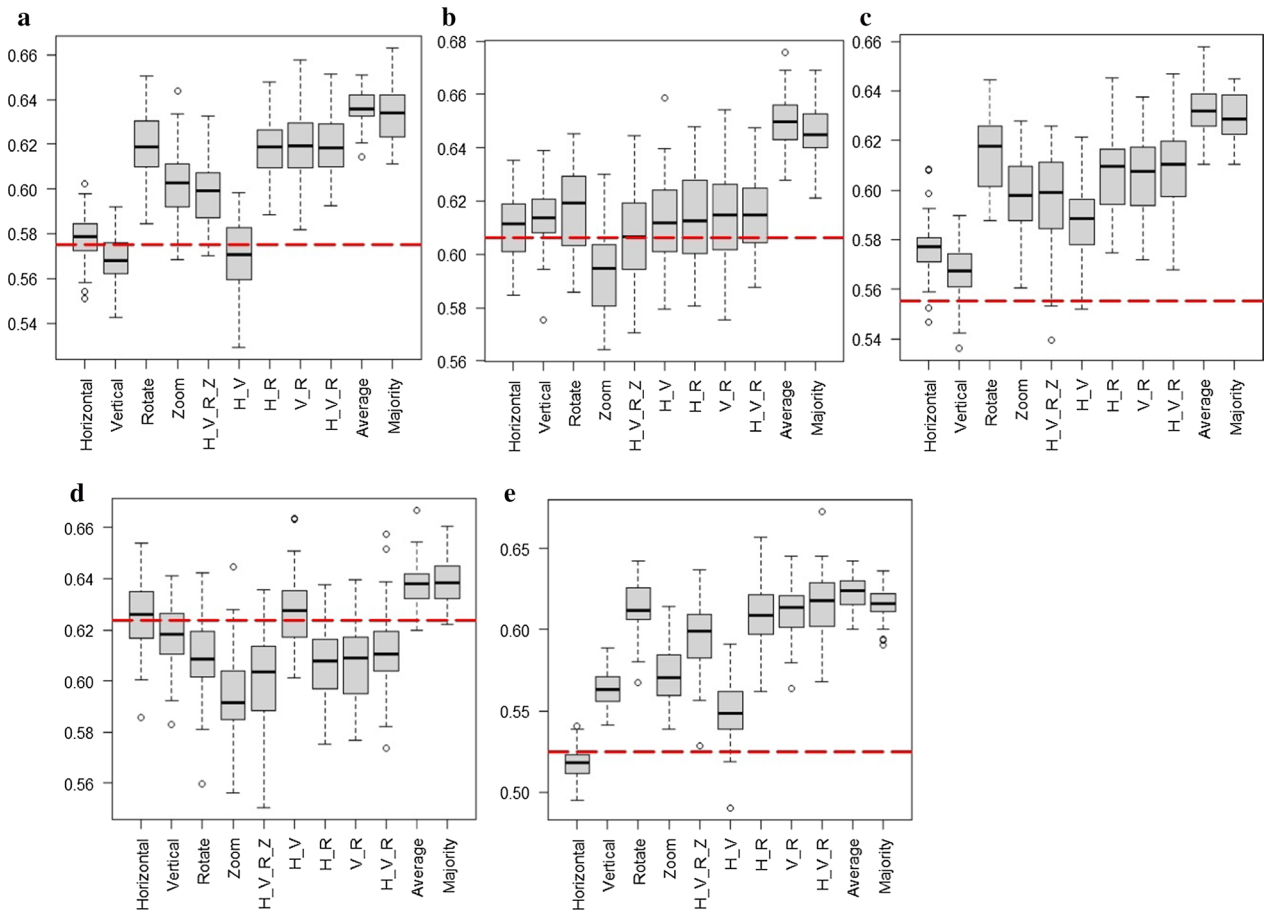

Fig. 6 This figure shows the Kappa scores distributions of the 50 experiments of each network for Wrist images. **a** Kappa scores of VGG19 network; **b** Kappa scores of InceptionV3 network; **c** Kappa scores of ResNet50 network; **d** Kappa scores of Xception network; **e** Kappa scores of DenseNet121 network. The red line represents the score of each network without TTA

Table 7 Kappa score with 95% C.I. of test time augmentation of Elbow images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	0.6078	0.6252	0.5908	0.6336	0.6123
Horizontal	0.6137 ± 0.30%	0.6027 ± 0.41%	0.5720 ± 0.38%	0.6438 ± 0.33%	0.6121 ± 0.35%
Vertical	0.6207 ± 0.30%	0.6244 ± 0.45%	0.5635 ± 0.47%	0.6587 ± 0.37%	0.6219 ± 0.41%
Rotate	0.6235 ± 0.49%	0.6253 ± 0.68%	0.6317 ± 0.65%	0.6665 ± 0.46%	0.6431 ± 0.56%
Zoom	0.5835 ± 0.60%	0.6146 ± 0.68%	0.6208 ± 0.71%	0.6181 ± 0.55%	0.6141 ± 0.64%
H_V_R_Z	0.6094 ± 0.47%	0.6067 ± 0.68%	0.6152 ± 0.71%	0.6573 ± 0.67%	0.6238 ± 0.55%
H_V	0.6181 ± 0.33%	0.6161 ± 0.63%	0.5586 ± 0.48%	0.6611 ± 0.52%	0.6158 ± 0.53%
H_R	0.6214 ± 0.44%	0.6376 ± 0.66%	0.6247 ± 0.66%	0.6653 ± 0.58%	0.6470 ± 0.54%
V_R	0.6205 ± 0.51%	0.6277 ± 0.55%	0.6277 ± 0.65%	0.6664 ± 0.49%	0.6494 ± 0.50%
H_V_R	0.6217 ± 0.42%	0.6363 ± 0.50%	0.6206 ± 0.65%	0.6668 ± 0.55%	0.6481 ± 0.60%
Average vote	0.6330 ± 0.28%	0.6810 ± 0.39%	0.6560 ± 0.33%	0.6937 ± 0.33%	0.6793 ± 0.31%
Majority vote	0.6308 ± 0.27%	0.6722 ± 0.37%	0.6473 ± 0.36%	0.6947 ± 0.35%	0.6748 ± 0.36%

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

augmentation techniques, with a Kappa score of 0.6494. The average vote and the majority vote scores were higher than both the original and nine different techniques. Overall, the average vote achieved the highest score, with a Kappa score of 0.6793, which increased 10.95% over the original method.

Overall, by comparing only the average vote and the majority vote scores to the original method, the most significant performance improvement was obtained by the ResNet50 network with a difference of 11.03% for the average vote and 9.56% for the majority vote. The lowest performance gain was returned by the VGG19 network, with a difference of 4.14% for the average vote and a difference of 3.77% for the majority vote. The majority vote method's average performance for all the networks was 8.14% and 8.91% for the average vote. The best performance among the five different networks without any TTA was achieved by the Xception network with a Kappa score of 0.6336, while the best score using the TTA methods was achieved by taking the majority vote of the nine different augmentation techniques of the Xception network with a Kappa score of 0.6947. Simultaneously, the lowest score among the five different networks was the score of the ResNet50, which was 0.5908. Figure 7 shows Kappa scores distributions of the 50 experiments of each network.

Hand images

The results of the nine different augmentation techniques with the average vote and the majority vote for the hand images are presented in Table 8. For the VGG19 network, the Kappa score of the model without any augmentation was 0.4032. The Kappa scores of the horizontal, vertical, zoom, and H_V augmentation techniques were similar to the original method. The rest of the augmentation techniques outperformed the original method. The highest Kappa score was achieved by the rotate method. Both the average vote and the majority vote were the highest among all the others. Overall, for the VGG19 network, the majority vote achieved the highest score with a Kappa score of 0.4542, representing an increase of 12.66% over the original score.

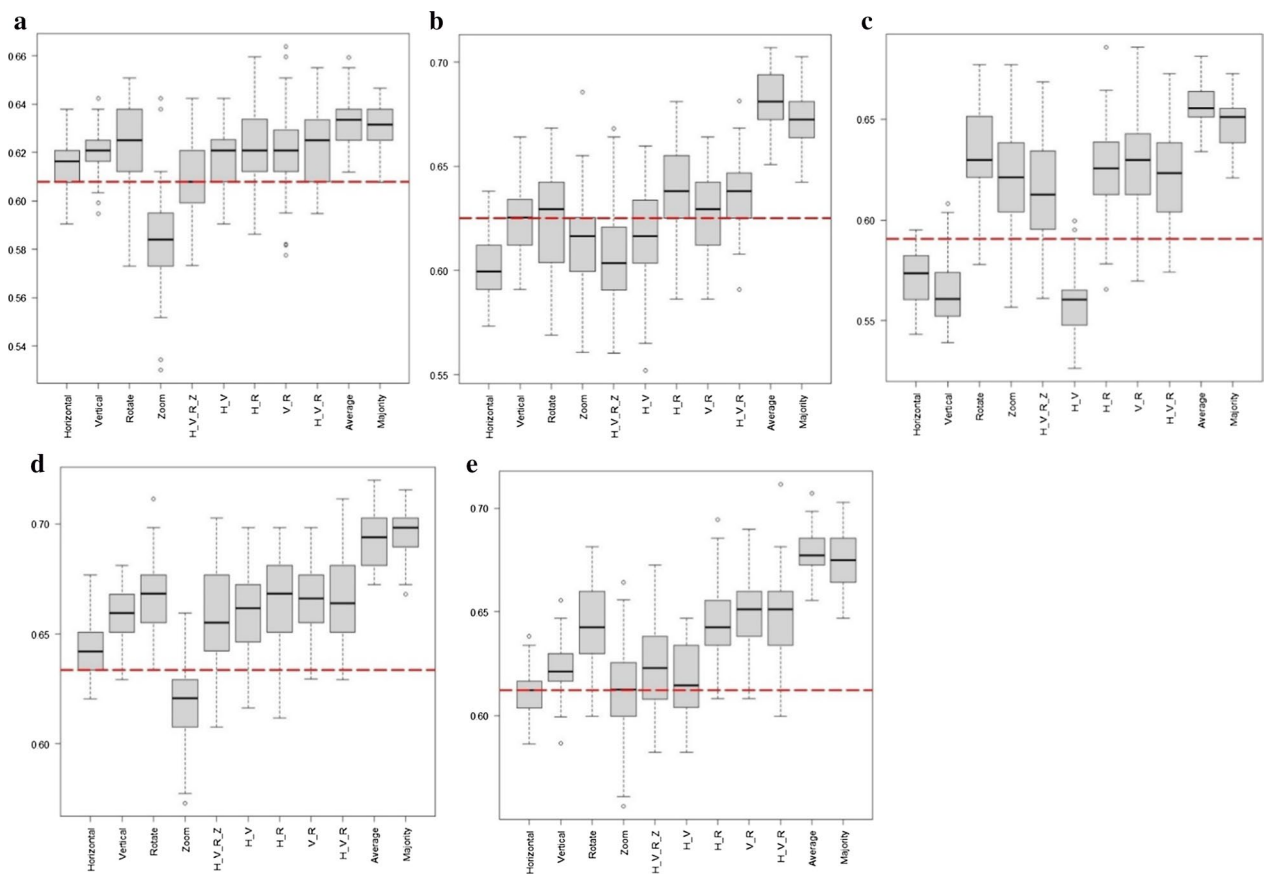

Fig. 7 This figure shows the Kappa scores distributions of the 50 experiments of each network for Elbow images. **a** Kappa scores of VGG19 network; **b** Kappa scores of InceptionV3 network; **c** Kappa scores of ResNet50 network; **d** Kappa scores of Xception network; **e** Kappa scores of DenseNet121 network. The red line represents the score of each network without TTA

Table 8 Kappa score of test time augmentation of Hand images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	0.4032	0.3762	0.3579	0.3741	0.3118
Horizontal	0.4022 ± 0.37%	0.3771 ± 0.40%	0.3685 ± 0.57%	0.3810 ± 0.39%	0.3175 ± 0.38%
Vertical	0.4076 ± 0.32%	0.3641 ± 0.39%	0.3676 ± 0.36%	0.4058 ± 0.40%	0.3064 ± 0.39%
Rotate	0.4488 ± 0.54%	0.3987 ± 0.65%	0.3785 ± 0.45%	0.4044 ± 0.56%	0.3995 ± 0.58%
Zoom	0.4067 ± 0.76%	0.3991 ± 0.74%	0.3599 ± 0.60%	0.3653 ± 0.67%	0.3664 ± 0.67%
H_V_R_Z	0.4244 ± 0.69%	0.4011 ± 0.65%	0.3654 ± 0.76%	0.3825 ± 0.67%	0.3942 ± 0.75%
H_V	0.4016 ± 0.52%	0.3675 ± 0.53%	0.3737 ± 0.62%	0.4106 ± 0.53%	0.3044 ± 0.63%
H_R	0.4466 ± 0.54%	0.4051 ± 0.57%	0.3789 ± 0.51%	0.4090 ± 0.50%	0.4077 ± 0.51%
V_R	0.4475 ± 0.56%	0.4140 ± 0.65%	0.3798 ± 0.54%	0.4066 ± 0.59%	0.4029 ± 0.61%
H_V_R	0.4462 ± 0.51%	0.4045 ± 0.62%	0.3858 ± 0.44%	0.4049 ± 0.47%	0.3981 ± 0.53%
Average vote	0.4539 ± 0.28%	0.4330 ± 0.35%	0.4018 ± 0.26%	0.4206 ± 0.28%	0.3916 ± 0.32%
Majority vote	0.4542 ± 0.35%	0.4274 ± 0.37%	0.3996 ± 0.30%	0.4164 ± 0.33%	0.3910 ± 0.30%

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

For the InceptionV3 network, the original Kappa score was 0.3762. Only the vertical method and the H_V method scored lower than the original method, while the rest of the methods outperformed the original method. The highest Kappa score was achieved by V_R with a Kappa score of 0.414. The average vote and the majority vote scored higher than the original method. Overall, for the InceptionV3 network, the best performance was achieved by using the average vote with a Kappa score of 0.4330, which represents an increase of 15.11% over the original method. For the ResNet50 network, the original method score was 0.3579. All the augmentation methods used yielded better results than the original method. The best performance among the nine different methods was obtained by the H_V_R method with a Kappa score of 0.3858. The average vote and the majority vote Kappa scores were the highest compared to all the experiments. Overall, for the ResNet50 network, the average vote achieved the highest score with a Kappa score of 0.4018, representing an increase of 12.27% over the original method.

For the Xception network, the original method score was 0.3741. Only the zoom augmentation method score was lower than the original method. The rest of the techniques outperformed the original method. The highest score was achieved by using the H_V method with a Kappa score of 0.4106. The average vote and the majority vote scores were higher than all the others. Concerning the Xception network, the average vote achieved the best Kappa score with a value of 0.4206, representing an increase of 12.43% over the original method. For the DenseNet121 network, the original model had a score of 0.3118. Only the vertical and the H_V methods had a score lower than the original method. The highest score among the nine different methods was achieved by the H_R method. Both the H_R and the V_R scores were higher than both the average vote and the majority vote. Overall, for the DenseNet121 network, the H_R method achieved the highest score with a Kappa of 0.4077, representing an increase of 30.76% over the original method.

By comparing only the average vote and the majority vote scores to the original method, the most significant performance improvement was produced by the DenseNet121 network with a difference of 25.60% for the average vote and 25.39% for the majority vote. The lowest performance gain was obtained by the Xception network, with a difference of 12.43% for the average vote and a difference of 11.30% for the majority vote. The majority vote method's average performance for all the networks was 14.92% and 15.60% for the average vote. The best performance among the five different networks without any TTA was achieved by the VGG19 network with a Kappa score of 0.4032. On the other hand, the best score using TTA was achieved by taking the majority vote of the augmentation techniques of the VGG19 network, with a Kappa score of 0.4542. Simultaneously, the lowest score among the five different networks was the score of the DenseNet121, which was 0.3118. Figure 8 shows Kappa scores distributions of the 50 experiments of each network.

Shoulder images

The results of the nine different augmentation techniques with the average vote and the majority vote for the shoulder images are presented in Table 9. For the VGG19 network, the original method score was 0.4357. All the augmentation techniques outperform the original score. The rotate augmentation technique achieved

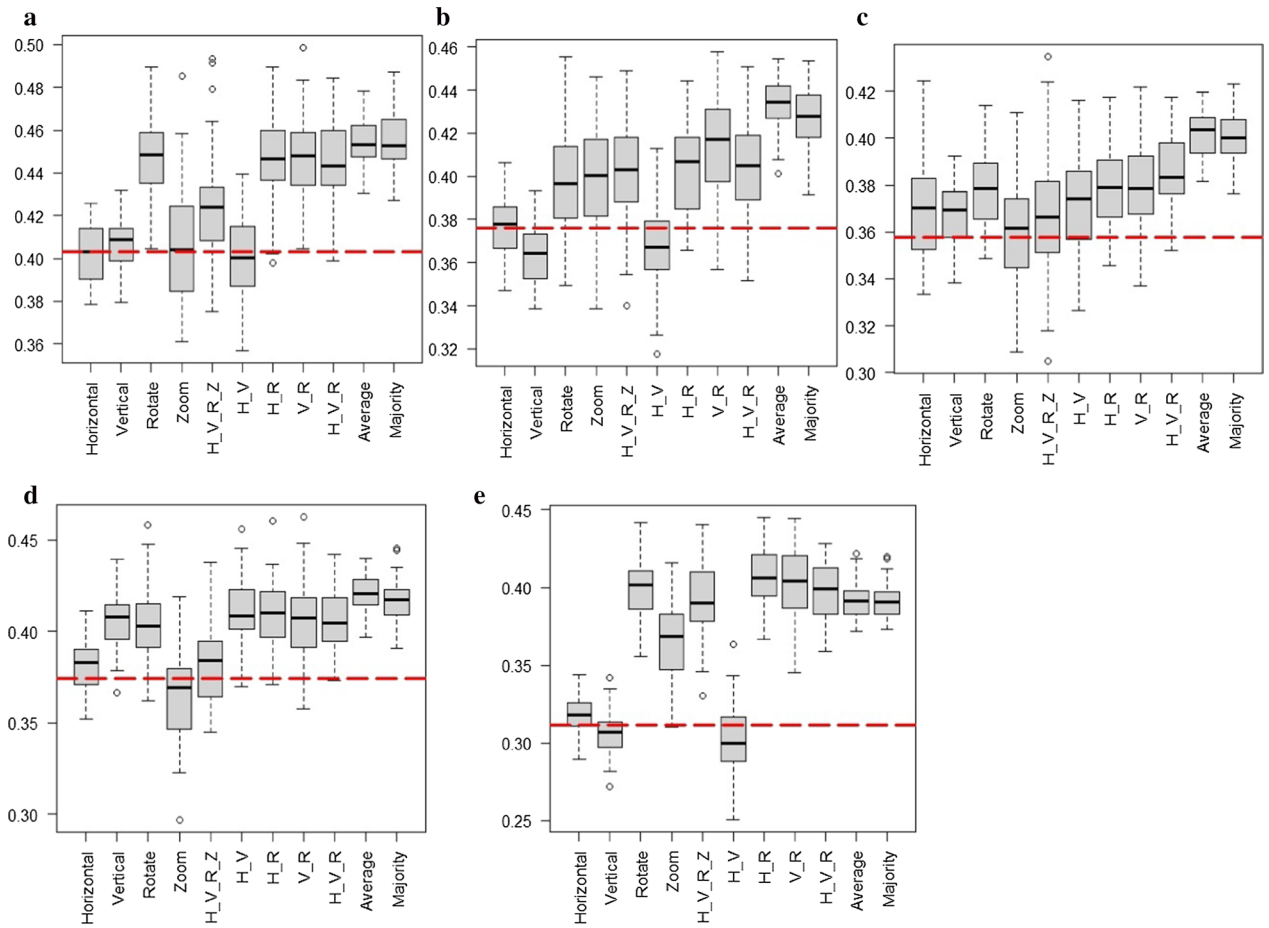

Fig. 8 This figure shows the Kappa scores distributions of the 50 experiments of each network for Hand images. **a** Kappa scores of VGG19 network; **b** Kappa scores of InceptionV3 network; **c** Kappa scores of ResNet50 network; **d** Kappa scores of Xception network; **e** Kappa scores of DenseNet121 network. The red line represents the score of each network without TTA

Table 9 Kappas score of test time augmentation of Shoulder images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	0.4357	0.3693	0.3203	0.4187	0.4013
Horizontal	0.4469 ± 0.36%	0.3492 ± 0.40%	0.3389 ± 0.41%	0.4066 ± 0.38%	0.4192 ± 0.37%
Vertical	0.4494 ± 0.42%	0.3872 ± 0.38%	0.3294 ± 0.46%	0.4367 ± 0.41%	0.4332 ± 0.43%
Rotate	0.4661 ± 0.65%	0.4814 ± 0.56%	0.4760 ± 0.61%	0.5056 ± 0.53%	0.4898 ± 0.56%
Zoom	0.4418 ± 0.58%	0.4675 ± 0.51%	0.4653 ± 0.58%	0.4895 ± 0.73%	0.4518 ± 0.59%
H_V_R_Z	0.4484 ± 0.73%	0.4703 ± 0.62%	0.4495 ± 0.67%	0.5020 ± 0.71%	0.4829 ± 0.66%
H_V	0.4531 ± 0.51%	0.3633 ± 0.54%	0.3354 ± 0.60%	0.4289 ± 0.50%	0.4397 ± 0.48%
H_R	0.4623 ± 0.52%	0.4802 ± 0.64%	0.4654 ± 0.64%	0.5024 ± 0.64%	0.4915 ± 0.61%
V_R	0.4657 ± 0.54%	0.4817 ± 0.67%	0.4672 ± 0.62%	0.5101 ± 0.60%	0.4940 ± 0.47%
H_V_R	0.4642 ± 0.73%	0.4831 ± 0.66%	0.4650 ± 0.74%	0.5066 ± 0.69%	0.4862 ± 0.61%
Average vote	0.4845 ± 0.32%	0.4977 ± 0.35%	0.4608 ± 0.43%	0.5221 ± 0.40%	0.5129 ± 0.36%
Majority vote	0.4825 ± 0.36%	0.4944 ± 0.40%	0.4693 ± 0.48%	0.5230 ± 0.51%	0.5015 ± 0.47%

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

the best Kappa score with a value of 0.4661. The average vote and the majority scores were higher than both the nine methods and the original method. Overall, for the VGG19 network, the best score was achieved by taking the average vote with a Kappa score of 0.4845, representing an increase of 11.19%.

For the InceptionV3 network, the original method score was 0.3693. The horizontal method and the H_V method scored lower than the original method. The rest of the techniques outperform the original network. The H_V_R augmentation technique achieved the highest score among the nine methods. The average vote and the majority vote scores were higher than all the other experiments. Overall, for the InceptionV3 network, the average vote achieved the highest score with a Kappa score of 0.4977, representing an increase of 34.78% over the original method.

For the ResNet50 network, the original method score was 0.3203. All the augmentation techniques outperformed the original method. The highest score was achieved by the rotate augmentation technique. The score achieved by the rotate method was even higher than both the average vote and the majority vote. Overall, for the ResNet50 network, the best score was achieved by the rotate augmentation technique with a Kappa score of 0.4761, representing an increase of 48.64% over the original method.

For the Xception network, the original method score was 0.4187. The horizontal augmentation technique was the only method that scored lower than the original method. All the other methods outperform the original method. The V_R augmentation technique achieved the highest Kappa score among the nine augmentation techniques with a value of 0.5101. Both the average vote and the majority vote scores were higher than all the other experiments. The majority vote achieved the highest Kappa score with a value of 0.5230, representing an increase of 24.91% over the original method. It is worth noting that the average vote score was lower than the majority vote with a Kappa score of 0.5221, which represents an increase of 24.70% over the original method. For the DenseNet121 network, the original method score was 0.4013. All the nine augmentation techniques outperformed the original method, with the V_R being the highest with a Kappa score of 0.4940. Both the average vote and the majority vote scores were higher than all the others. Overall, for the DenseNet121 network, the average vote achieved the best score with a Kappa score of 0.5129, representing an increase of 27.80% over the original method.

Overall, by comparing only the average vote and the majority vote scores to the original method, the highest performance gain was achieved by the ResNet50 network with a difference of 43.86% for the average vote and 46.51% for the majority vote. The lowest performance gain was produced by the VGG19 network, with a difference of 11.19% for the average vote and a difference of 10.75% for the majority vote. The majority vote method's average performance for all the networks was 28.20% and 28.47% for the average vote. The best performance among the five different networks without any TTA was achieved by the VGG19 network with a Kappa score of 0.4357, while the best score using the TTA methods was achieved by taking the majority vote of the nine different augmentation techniques of the Xception network with a Kappa score of 0.5230. Simultaneously, the lowest score among the five different networks was the score of the ResNet50, which was 0.3203. Figure 9 shows Kappa scores distributions of the 50 experiments of each network.

Discussion

The problem of accurately classifying musculoskeletal images in the ER is of extreme relevance. The presence of an automatic classifier in the ER can significantly reduce the errors made [2]. However, due to several reasons like the limited availability of large datasets and the quality of others, the performance of such a classifier needs to improve [24, 26, 27]. In this paper, we investigated the role of TTA to understand whether it can increase the classifier performance without increasing the computational effort needed. The remaining part of this section discusses some insights into our results.

Ensemble learning has achieved superior performance compared to single models in several studies [29–31]. Ensemble learning can be defined as using more than one classifier for prediction [32]. Thus, taking the average vote or the majority vote of the augmentation techniques can be considered as a particular example of ensemble learning. The average vote score was higher for the finger dataset than all the scores except for the VGG19 network and the InceptionV3 network. For the humerus dataset, the average vote score was higher than all the scores except for the majority vote of the VGG19 network.

For the forearm dataset, the majority vote score was higher than all the scores except for the InceptionV3 and the Xception networks. For both the wrist dataset and the elbow dataset, the average vote produced the highest score except for the Xception network, where the majority vote outperformed the average vote. For the hand dataset, the average vote returned the highest score except for the VGG19 network, where the majority vote outperformed the average vote. For the shoulder dataset, the average vote was better than the majority vote except

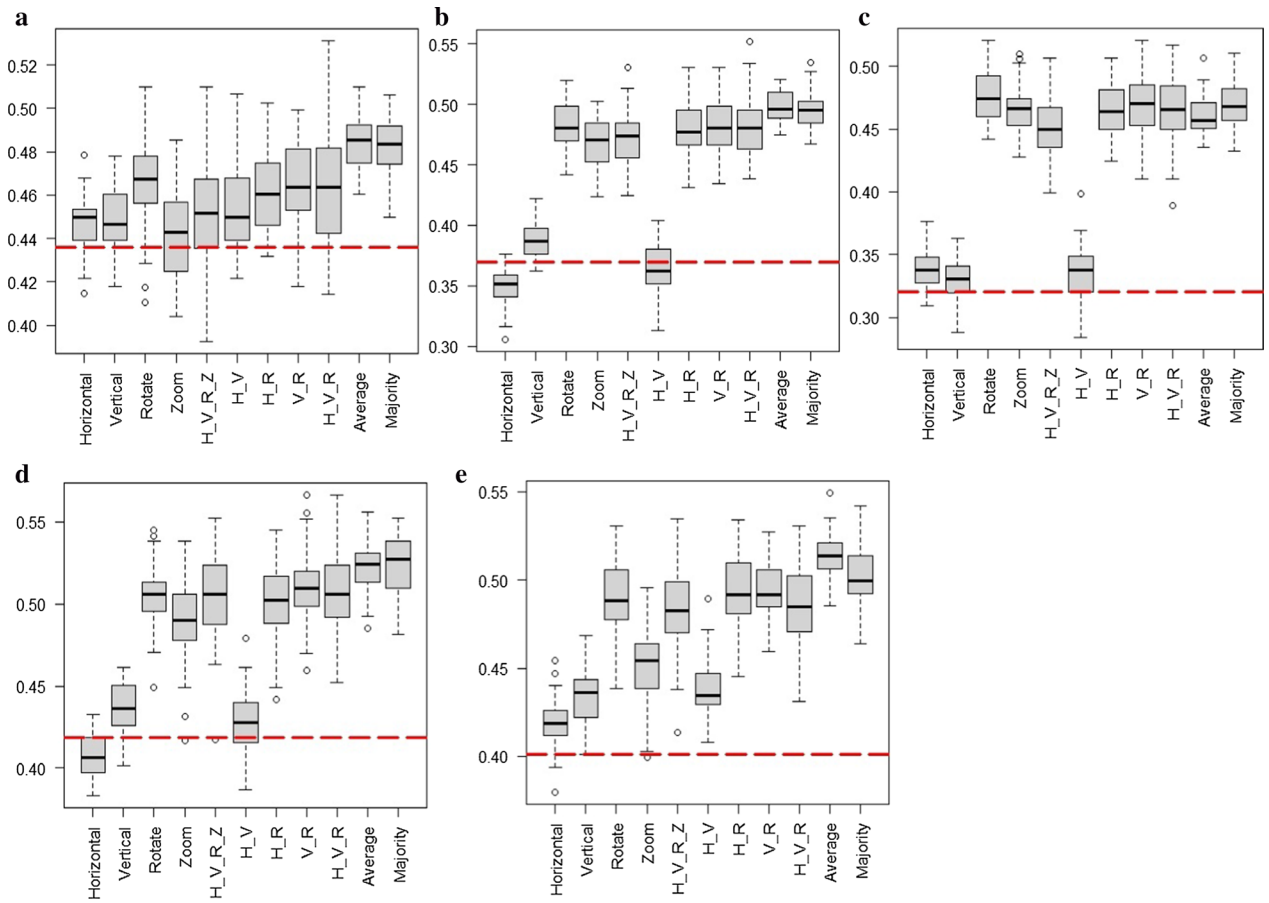

Fig. 9 This figure shows the Kappa scores distributions of the 50 experiments of each network for Shoulder images. **a** Kappa scores of VGG19 network; **b** Kappa scores of InceptionV3 network; **c** Kappa scores of ResNet50 network; **d** Kappa scores of Xception network; **e** Kappa scores of DenseNet121 network. The red line represents the score of each network without TTA

Table 10 Percentages of increase by using the TTA method compared to the original method

Dataset	Average vote (%)	Majority vote (%)
FINGER	22.87	21.53
HUMERUS	9.75	9.56
FOREARM	3.48	3.60
WRIST	10.48	9.96
ELBOW	8.91	8.14
HAND	15.60	14.92
SHOULDER	28.47	28.20

for the ResNet50 network and the Xception network. All in all, we can conclude that taking the average vote can be considered the best option compared to single models or the majority vote. Table 10 shows the average increase between the original method, the average vote, and the majority vote.

The horizontal flipping technique is a popular geometric augmentation technique that was used in many computer vision studies. In this study, we applied a random horizontal flipping to the images. The results obtained by horizontal flipping varied significantly between different datasets. For the finger dataset, its score was lower than the original method without TTA for all the networks except for the Xception network (an increase of 3.05% compared to the original method). For the Humerus images, the score of the model without TTA was higher than the horizontal flipping for all the networks except for the VGG19 network (an increase of 3.40% compared to the original method) and the Xception network (an increase of 2.58% compared to the original method). However, for both the

forearm dataset and the wrist dataset, the horizontal flipping score was higher than the original score for all the networks except for the DenseNet121 network. For the elbow dataset, the horizontal flipping score was higher than the original method only for the VGG19 network (an increase of 0.96% compared to the original method) and the Xception network (an increase of 1.61% compared to the original method). For the hand dataset, horizontal flipping was neither beneficial nor detrimental, and the results were approximately the same as the original method. For the shoulder dataset, horizontal flipping results were poorer than the original method for the InceptionV3 network and Xception network. Overall, horizontal flipping did have a higher score than the original method 20 out of 35 times.

The vertical flipping technique was applied, just like the horizontal flipping technique. For the shoulder dataset, the vertical flipping technique achieved higher results than the original method for all the networks, with an average increase of 4.62% compared to the original method. For the humerus dataset and the forearm dataset, the vertical flipping score was lower than the original method for both the VGG19 network and the Xception network. For the Finger dataset, the original method's score was higher than the flipping for all the networks except for the Xception network (an increase of 5.03% compared to the original method) and the DenseNet121 network (an increase of 7.04% compared to the original method). For the wrist dataset, the original method for the VGG19 network and the Xception network was higher than the flipping technique. For the elbow dataset, the score of the original method for both the InceptionV3 network and the ResNet50 network was higher than the flipping technique. The original method for the InceptionV3 network and the DenseNet121 is higher than the hand dataset's vertical flipping. Overall, the vertical flipping technique did have a higher score than the original method 23 out of 35 times.

Concerning the zooming augmentation technique, in this study, we applied a random 40% zooming. For the shoulder dataset and the finger dataset, the zooming score was significantly higher than the original method's score. For the forearm dataset, the results were completely the opposite with respect to both the shoulder dataset and the finger dataset, where the original method score was higher than the zooming technique for all the networks. For the humerus dataset, the zooming score was higher than the original method for all the networks except for the VGG19 network. For the wrist dataset, the results of the original method for both the InceptionV3 network and the Xception network were better than the zooming technique. For the elbow dataset, only the ResNet50 network and the DenseNet121 network achieved higher results than the original method. For the hand dataset, all the networks' scores were higher than the original method except for the Xception network. Overall, the zooming technique resulted in a higher score than the original method 23 out of 35 times.

Concerning the rotation technique, we applied random 180° rotations of the images. For the finger dataset, the elbow dataset, the hand dataset, and the shoulder dataset rotation achieved outstanding results, where the resulting score was always higher than the original model score. For the humerus dataset, the rotation score was higher than the original model, except for the InceptionV3 network, where the score was slightly lower than the original score. For the forearm dataset, the rotation was lower than the original score for all networks except for the InceptionV3 network. For the wrist dataset, the rotation was higher than the original model with a high margin except for the Xception network. Overall, the rotation technique produced a higher score than the original method 29 out of 35 times.

The confidence interval (CI) can be an indication of the stability of the algorithm being used. In this study, we used a 95% CI calculated over 50 experiments of every network. We noted that all the CIs were below 1% on average, which is a good indication of the robustness of the TTA results.

An exciting finding is that the TTA significantly impacted models with a low score rather than the models with high scores. For instance, in the finger dataset, the vast impact of TTA (29.94% increase over the original score) was observed in the InceptionV3 network that has the smallest Kappa score compared to the other considered networks. The same phenomenon was also observed in the humerus dataset, for the DensetNet121 network (13.72% increase over the original score); in the wrist dataset, for the DenseNet121 network (18.54% increase); in the elbow dataset, for the ResNet50 network (11.03% increase); in the hand dataset, for the DenseNet121 network (25.60% increase); and in the shoulder dataset, for the ResNet50 network (43.86% increase for the average vote and 46.51% for the majority vote).

All in all, across the different datasets and CNNs, TTA allows obtaining better performance than the traditional method (i.e., without TTA). This result can be explained by looking at the TTA as an ensemble built during the testing phase. In other words, the possibility of obtaining the final prediction by combining the model's predictions over different augmentation strategies can correct the error produced by the model when it has to classify a

single image without considering its transformations. The idea is somehow similar to ensemble learning, but with an important difference: in ensemble models, the predictions of different weak learners are combined to provide a unique prediction for a specific observation. In this case, the same network builds its prediction (obtained as averaging or majority vote) by considering different transformations of the same image. Thus, while we can see some analogy with ensemble learning, we remark that the method is different and takes place only during the testing phase.

Of course, when considering only a single augmentation method, the results can be worse than the original method because the augmentation technique is not suitable for the specific task at hand. An example of this situation is a computer vision task where we must discriminate images of digits 6 and 9. In this situation, we cannot expect a vertical flipping to provide better performance than the baseline method. For this reason, it is fundamental to consider the combination of several augmentation techniques.

Using TTA did not increase the computational cost during the training phase. The reason is that the TTA will run after training the model, which means that TTA will be used post-processing. We investigated the effect of TTA on computation cost by calculating the running time of every model with and without TTA. For the sake of readability, all the tables with the computational time can be found in the Annex. As one can see from the tables in the Annex, TTA requires more time to evaluate the whole test set. However, the additional computational time is negligible if we consider the beneficial effect of TTA on the performance of the considered CNNs.

Conclusions

This work presented an extensive investigation of the usage of TTA on the musculoskeletal X-Ray images dataset. The MURA dataset consists of 40,005 images of seven different upper extremities. The dataset is considered very challenging because of some datasets' small size and the imbalance in others. Nine augmentation techniques were studied: namely, rotation; zooming; horizontal flipping; vertical flipping; horizontal flip with vertical flip (H_V); horizontal flip with rotation (H_R); vertical flip with rotation (V_R); horizontal flip, vertical flip, and rotation (H_V_R); horizontal flip, vertical flip, rotation, and zooming (H_V_R_Z). Two ensemble methods were also tested: the average vote of the nine augmentation techniques and the majority vote. It was observed that taking the nine augmentation techniques' average vote produced the best performance. Our results show that TTA can increase the classifier's performance without adding any computational cost during training. For our future work, we plan to investigate the role of TTA on other medical domains, especially on 3D images for MRI images or CT scan images, to see its effect.

Appendix

See Tables 11, 12, 13, 14, 15, 16, 17.

Table 11 Computational time of test time augmentation of Finger images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	4.20 s	7.10 s	6.20 s	5.55 s	7.50 s
Horizontal	4.43 s ± 0.20	7.60 s ± 0.37	6.33 s ± 0.37	5.91 s ± 0.43	7.71 s ± 0.28
Vertical	4.61 s ± 0.42	7.63 s ± 0.34	6.53 s ± 0.48	6.06 s ± 0.35	7.85 s ± 0.37
Rotate	5.58 s ± 0.37	8.71 s ± 0.28	7.66 s ± 0.65	7.47 s ± 0.40	8.76 s ± 0.21
Zoom	5.92 s ± 0.53	8.83 s ± 0.27	7.65 s ± 0.47	7.38 s ± 0.48	8.87 s ± 0.33
H_V_R_Z	6.18 s ± 0.51	11.64 s ± 0.64	7.08 s ± 0.34	6.67 s ± 0.47	8.60 s ± 0.31
H_V	4.57 s ± 0.33	7.65 s ± 0.34	6.25 s ± 0.46	5.77 s ± 0.39	7.83 s ± 0.21
H_R	5.82 s ± 0.40	8.98 s ± 0.38	7.47 s ± 0.49	7.30 s ± 0.41	8.74 s ± 0.26
V_R	5.83 s ± 0.41	9.05 s ± 0.44	8.81 s ± 0.66	7.69 s ± 0.52	11.74 s ± 0.52
H_V_R	5.63 s ± 0.58	8.52 s ± 0.31	7.40 s ± 0.70	6.69 s ± 0.36	8.76 s ± 0.27

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Table 12 Computational time of test time augmentation of Humerus images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	2.50 s	4.50 s	4.25 s	3.10 s	4.75 s
Horizontal	2.67 s ± 0.11	4.62 s ± 0.20	4.35 s ± 0.23	3.25 s ± 0.11	5.02 s ± 0.24
Vertical	2.64 s ± 0.08	4.62 s ± 0.16	4.49 s ± 0.24	3.32 s ± 0.22	4.90 s ± 0.20
Rotate	3.40 s ± 0.13	5.30 s ± 0.16	5.33 s ± 0.30	3.96 s ± 0.19	5.65 s ± 0.29
Zoom	3.37 s ± 0.14	5.32 s ± 0.28	5.66 s ± 0.50	4.02 s ± 0.20	5.70 s ± 0.32
H_V_R_Z	3.97 s ± 0.38	8.25 s ± 0.60	5 s ± 0.38	3.87 s ± 0.11	5.49 s ± 0.24
H_V	2.68 s ± 0.09	4.75 s ± 0.24	4.35 s ± 0.25	3.33 s ± 0.12	4.90 s ± 0.21
H_R	3.45 s ± 0.10	5.16 s ± 0.15	5.39 s ± 0.35	3.95 s ± 0.19	5.64 s ± 0.26
V_R	3.54 s ± 0.12	5.30 s ± 0.33	6.12 s ± 0.29	4.98 s ± 0.27	8.52 s ± 0.45
H_V_R	3.42 s ± 0.13	5.24 s ± 0.25	4.90 s ± 0.25	3.83 s ± 0.15	5.53 s ± 0.21

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Table 13 Computational time of test time augmentation of Forearm images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	2.80 s	4.50 s	4.20 s	3.0 s	5.0 s
Horizontal	3.07 s ± 0.15	4.78 s ± 0.19	4.96 s ± 0.68	3.33 s ± 0.16	5.09 s ± 0.21
Vertical	3.07 s ± 0.18	4.80 s ± 0.21	4.76 s ± 0.53	3.24 s ± 0.10	5.13 s ± 0.23
Rotate	3.84 s ± 0.13	5.47 s ± 0.17	5.43 s ± 0.49	3.79 s ± 0.13	5.75 s ± 0.18
Zoom	3.96 s ± 0.16	5.39 s ± 0.17	5.27 s ± 0.44	3.87 s ± 0.16	5.75 s ± 0.20
H_V_R_Z	4.14 s ± 0.33	7.90 s ± 0.52	5.07 s ± 0.25	3.79 s ± 0.13	5.69 s ± 0.19
H_V	2.96 s ± 0.13	4.83 s ± 0.17	4.82 s ± 0.57	3.28 s ± 0.13	5.17 s ± 0.21
H_R	3.94 s ± 0.15	5.46 s ± 0.20	5.64 s ± 0.59	3.81 s ± 0.12	5.80 s ± 0.29
V_R	3.92 s ± 0.20	5.47 s ± 0.22	6.23 s ± 0.38	4.95 s ± 0.28	8.97 s ± 0.61
H_V_R	3.86 s ± 0.18	5.44 s ± 0.20	5.36 s ± 0.55	3.81 s ± 0.11	5.75 s ± 0.22

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Table 14 Computational time of test time augmentation of Wrist images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	7.0 s	13.0 s	9.0 s	7.25 s	11.5 s
Horizontal	7.38 s ± 0.62	13.68 s ± 1.09	9.40 s ± 0.87	7.46 s ± 0.35	12.30 s ± 0.58
Vertical	7.04 s ± 0.37	14.11 s ± 1.79	9.47 s ± 0.93	7.63 s ± 0.38	12.60 s ± 0.42
Rotate	9.05 s ± 0.38	15.95 s ± 1.71	11.03 s ± 0.89	9.17 s ± 0.58	14.10 s ± 0.54
Zoom	9.17 s ± 0.50	15.53 s ± 1.07	11.52 s ± 1.05	9.38 s ± 0.62	14.66 s ± 0.63
H_V_R_Z	8.64 s ± 0.53	15.81 s ± 1.02	10.06 s ± 0.53	8.61 s ± 0.24	12.77 s ± 0.44
H_V	7.03 s ± 0.51	13.57 s ± 1.26	9.24 s ± 0.79	7.35 s ± 0.34	11.92 s ± 0.60
H_R	9.23 s ± 0.38	15.60 s ± 1.04	11.35 s ± 0.87	9.08 s ± 0.54	13.94 s ± 0.53
V_R	9.09 s ± 0.32	15.78 s ± 1.23	11.36 s ± 0.44	9.48 s ± 0.21	16.13 s ± 0.70
H_V_R	8.61 s ± 0.42	13.83 s ± 1.07	10.52 s ± 0.57	8.72 s ± 0.39	13.36 s ± 0.45

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Table 15 Computational time of test time augmentation of Elbow images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	4.0 s	8.25 s	6.35 s	5.35 s	8.15 s
Horizontal	4.86 s ± 0.34	8.40 s ± 0.79	6.79 s ± 0.48	5.72 s ± 0.28	8.39 s ± 0.38
Vertical	4.88 s ± 0.31	8.46 s ± 0.93	6.82 s ± 0.54	5.76 s ± 0.37	8.62 s ± 0.33
Rotate	6.05 s ± 0.44	9.74 s ± 0.89	8.11 s ± 0.66	6.86 s ± 0.31	9.86 s ± 0.49
Zoom	6.03 s ± 0.35	10.29 s ± 1.02	8.15 s ± 0.70	6.84 s ± 0.47	10.33 s ± 0.75
H_V_R_Z	6.07 s ± 0.44	11.40 s ± 0.88	7.58 s ± 0.57	6.46 s ± 0.30	9.06 s ± 0.31
H_V	4.87 s ± 0.38	8.13 s ± 0.66	6.52 s ± 0.39	5.57 s ± 0.31	8.26 s ± 0.34
H_R	6.23 s ± 0.38	10.21 s ± 0.96	7.91 s ± 0.75	6.70 s ± 0.44	9.45 s ± 0.30
V_R	6.26 s ± 0.34	10.32 s ± 1.00	8.78 s ± 0.53	7.36 s ± 0.25	12.59 s ± 1.11
H_V_R	6.08 s ± 0.41	9.04 s ± 0.64	7.43 s ± 0.34	6.46 s ± 0.32	9.23 s ± 0.32

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Table 16 Computational time of test time augmentation of Hand images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	4.80 s	9.10 s	6.5 s	5.0 s	9.0 s
Horizontal	4.98 s ± 0.46	9.69 s ± 0.53	6.78 s ± 0.37	5.20 s ± 0.29	9.08 s ± 0.89
Vertical	5.10 s ± 0.44	10.01 s ± 0.61	7.07 s ± 0.51	5.16 s ± 0.35	9.42 s ± 1.11
Rotate	6.49 s ± 0.63	11.40 s ± 0.56	8.67 s ± 0.55	6.29 s ± 0.39	10.53 s ± 0.77
Zoom	6.57 s ± 0.59	11.36 s ± 0.64	8.90 s ± 0.46	6.43 s ± 0.38	10.61 s ± 1.02
H_V_R_Z	6.26 s ± 0.56	12.43 s ± 0.55	7.39 s ± 0.30	6.07 s ± 0.21	9.46 s ± 0.62
H_V	5.13 s ± 0.50	9.59 s ± 0.55	6.66 s ± 0.34	5.19 s ± 0.27	9.15 s ± 1.23
H_R	6.53 s ± 0.62	11.53 s ± 0.55	8.29 s ± 0.54	6.34 s ± 0.29	10.51 s ± 1.05
V_R	6.67 s ± 0.68	11.71 s ± 0.81	9.04 s ± 0.40	7.08 s ± 0.22	12.67 s ± 0.74
H_V_R	6.35 s ± 0.75	10.23 s ± 0.44	7.53 s ± 0.39	6.20 s ± 0.35	9.66 s ± 0.61

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Table 17 Computational time of test time augmentation of Shoulder images (± C.I.)

Technique	VGG19	InceptionV3	ResNet50	Xception	DenseNet121
Without TTA	6.0 s	9.50 s	7.5 s	6.25 s	10.50 s
Horizontal	6.46 s ± 0.43	10.08 s ± 0.70	7.97 s ± 0.34	6.46 s ± 0.32	11.02 s ± 0.49
Vertical	6.19 s ± 0.54	9.68 s ± 0.75	8.07 s ± 0.36	6.30 s ± 0.29	11.30 s ± 0.70
Rotate	8.35 s ± 0.60	11.60 s ± 1.01	9.31 s ± 0.40	7.59 s ± 0.44	13.49 s ± 0.93
Zoom	8.14 s ± 0.61	11.11 s ± 0.73	9.20 s ± 0.41	7.73 s ± 0.50	13.43 s ± 0.67
H_V_R_Z	7.82 s ± 0.57	10.67 s ± 0.71	8.73 s ± 0.38	7.51 s ± 0.30	11.74 s ± 0.50
H_V	6.43 s ± 0.46	9.70 s ± 0.69	8.17 s ± 0.33	6.43 s ± 0.27	10.81 s ± 0.39
H_R	8.26 s ± 0.64	10.84 s ± 0.68	9.40 s ± 0.44	7.67 s ± 0.43	13.23 s ± 0.86
V_R	8.56 s ± 0.69	12.92 s ± 1.01	10.33 s ± 0.88	8.47 s ± 0.34	15.31 s ± 0.97
H_V_R	7.57 s ± 0.56	10.86 s ± 0.71	8.93 s ± 0.44	7.47 s ± 0.30	12.08 s ± 0.43

Where H_V, stands for the horizontal flip with vertical flip; H_R, for the horizontal flip with rotation; V_R, for the vertical flip with rotation; H_V_R, stands for the horizontal flip, vertical flip, and rotation; and H_V_R_Z, stands for combining all four methods, horizontal flip, vertical flip, rotation, and zooming

Acknowledgements
This work was supported by national funds through FCT (Fundação para a Ciência e a Tecnologia) by the project GADgET (DSAIPA/DS/0022/2018) and the financial support from the Slovenian Research Agency (research core Funding No. P5-0410).

Data availability
The MURA dataset underlying this study is a publicly available dataset available from http://arxiv.org/abs/1712.06957. The authors do not have special access privileges to these data and confirm that interested researchers may access them via the link provided.

Declarations

Conflict of interest
The authors declare that there is no conflict of interest.

Received: 20 December 2020 Accepted: 19 July 2021
Published online: 31 July 2021

References

1. Hallas P, Ellingsen T. Errors in fracture diagnoses in the emergency department - characteristics of patients and diurnal variation. BMC Emerg Med. 2006. https://doi.org/10.1186/1471-227X-6-4.
2. Lindsey R, et al. Deep neural network improves fracture detection by clinicians. Proc Natl Acad Sci. 2018;115(45):11591–6. https://doi.org/10.1073/pnas.1806905115.
3. Pan S, Yang Q. A survey on transfer learning. IEEE Trans Knowl Data Eng. 2010;22:1345–59. https://doi.org/10.1109/TKDE.2009.191.
4. Kandel I, Castelli M. How deeply to fine-tune a convolutional neural network: a case study using a histopathology dataset. Appl Sci. 2020;10(10):3359. https://doi.org/10.3390/APP10103359.
5. Sharma S, Mehra DR. Breast cancer histology images classification: training from scratch or transfer learning? ICT Express. 2018. https://doi.org/10.1016/j.icte.2018.10.007.
6. Tajbakhsh N, et al. Convolutional neural networks for medical image analysis: full training or fine tuning? IEEE Trans Med Imaging. 2016;35(5):1299–312. https://doi.org/10.1109/TMI.2016.2535302.
7. Shorten C, Khoshgoftaar TM. A survey on image data augmentation for deep learning. J Big Data. 2019. https://doi.org/10.1186/s40537-019-0197-0.
8. Mylonas A, et al. A deep learning framework for automatic detection of arbitrarily shaped fiducial markers in intrafraction fluoroscopic images. Med Phys. 2019;46(5):2286–97. https://doi.org/10.1002/mp.13519.
9. Ahn JM, Kim S, Ahn K-S, Cho S-H, Lee KB, Kim US. A deep learning model for the detection of both advanced and early glaucoma using fundus photography. PLoS ONE. 2018;13(11):e0207982. https://doi.org/10.1371/journal.pone.0207982.
10. Chen Q, Hu S, Long P, Lu F, Shi Y, Li Y. A Transfer Learning Approach for Malignant Prostate Lesion Detection on Multiparametric MRI. Technol Cancer Res Treat. 2019. https://doi.org/10.1177/1533033819858363.
11. Gong H, et al. A deep learning- and partial least square regression-based model observer for a low-contrast lesion detection task in CT. Med Phys. 2019;46(5):2052–63. https://doi.org/10.1002/mp.13500.
12. Pang S, Yu Z, Orgun MA. A novel end-to-end classifier using domain transferred deep convolutional neural networks for biomedical images. Comput Methods Programs Biomed. 2017;140:283–93. https://doi.org/10.1016/j.cmpb.2016.12.019.
13. Rane C, Mehrotra R, Bhattacharyya S, Sharma M, Bhattacharya M. A novel attention fusion network-based framework to ensemble the predictions of CNNs for lymph node metastasis detection. J Supercomput. 2020. https://doi.org/10.1007/s11227-020-03432-6.
14. Wang G, Li W, Aertsen M, Deprest J, Ourselin S, Vercauteren T. Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. Neurocomputing. 2019;338:34–45. https://doi.org/10.1016/j.neucom.2019.01.103.
15. Amiri M, Brooks R, Behboodi B, Rivaz H. Two-stage ultrasound image segmentation using U-Net and test time augmentation. Int J Comput Assist Radiol Surg. 2020;15(6):981–8. https://doi.org/10.1007/s11548-020-02158-3.
16. Sigurthorsdottir H, Van Zaen J, Delgado-Gonzalo R, Lemay M. ECG classification with a convolutional recurrent neural network. 2020. http://arxiv.org/abs/2009.13320. Accessed 15 Nov 2020.
17. Wang G, Li W, Ourselin S, Vercauteren T. Automatic brain tumor segmentation using convolutional neural networks with test-time augmentation BT—brainlesion: glioma, multiple sclerosis, stroke and traumatic brain injuries. 2019, p. 61–72.
18. Simonyan K, Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2014. http://arxiv.org/abs/1409.1556.
19. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna ZB. Rethinking the inception architecture for computer vision. In: IEEE conference on computer vision and pattern recognition (CVPR). 2016. p. 2818–2826. https://doi.org/10.1109/CVPR.2016.308.
20. He K, Zhang X, Ren S, Sun J. "Deep residual learning for image recognition", in. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016;2016:770–8. https://doi.org/10.1109/CVPR.2016.90.
21. Chollet F. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
22. Huang G, Liu Z, Van der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017. pp. 2261–2269.
23. Shanmugam D, Blalock D, Balakrishnan G, Guttag J. When and why test-time augmentation works. 2020. http://arxiv.org/abs/2011.11156.
24. Rajpurkar P, Irvin J, Bagul A, Ding DY, Duan T, Mehta H, Yang BJ, Zhu K, Laird D, Ball RL, et al. MURA: Large Dataset for Abnormality Detection in Musculoskeletal Radiographs. 2017. http://arxiv.org/abs/1712.06957
25. Cohen J. A coefficient of agreement for nominal scales. Educ Psychol Measur. 1960;20(1):37–46. https://doi.org/10.1177/001316446002000104.
26. Chada G. Machine learning models for abnormality detection in musculoskeletal radiographs. Reports. 2019;2:26. https://doi.org/10.3390/reports2040026.
27. Kandel I, Castelli M, Popovič A. Musculoskeletal images classification for detection of fractures using transfer learning. J Imaging. 2020. https://doi.org/10.3390/jimaging6110127.
28. Kingma D, Ba J. Adam: A Method for Stochastic Optimization. In: Proceedings of the International Conference on Learning Representations (ICLR), Banff, AB, Canada, 14–16 April 2014.
29. Deng L, Platt J. Ensemble deep learning for speech recognition. In: Proc. interspeech, 2014. https://www.microsoft.com/en-us/research/publication/ensemble-deep-learning-for-speech-recognition/.
30. Zilly J, Buhmann JM, Mahapatra D. Glaucoma detection using entropy sampling and ensemble learning for automatic optic cup and disc segmentation. Comput Med Imaging Graph. 2017;55:28–41. https://doi.org/10.1016/j.compmedimag.2016.07.012.
31. Potes C, Parvaneh S, Rahman A, Conroy B. Ensemble of feature-based and deep learning-based classifiers for detection of abnormal heart sounds. In: 2016 Computing in cardiology conference (CinC). 2016. p. 621–624
32. Dietterich TG. Ensemble methods in machine learning BT—multiple classifier systems. 2000. P. 1–15.