Does Prior Knowledge Affect CNN and LSTM Models on Classifying Improper Sitting Postures?

GEORGIOS VALTAS, University of Twente, The Netherlands

Posture can be defined as the physiological position held by a body when it is maintaining a stationary stance such as sitting or standing. Holding a correct posture implies that the individual maintains a stance in a position that does not exert excess strain on the body and maintains the straight and natural curve of the spine [1]. Lack of having a proper posture can impose a negative impact over extended periods of time such as in the office space. In this study, 3 types of Neural Network models were evaluated and tested on images of individuals holding 3 different postures: leaning to the left, to the right and sitting straight. The 3 models: A simple CNN model was compared to 2 models with transfer learning, a CNN and an LSTM model, with the intention of measuring the importance of prior knowledge in models. Results showed that with a dataset of 2751 images, prior knowledge did not affect the models 99.7% vs 99.7% (with prior knowledge) yet with a smaller dataset of 540 images the model with prior knowledge performed better by 13.3% better. While determinants such as the background and types of clothing can affect the models, a variation of the cosine pose similarity equation was found to aid the models' accuracy.

Additional Key Words and Phrases: Posture, Pose Recognition, MoveNet, Inception, CNN, LSTM, RNN, Cosine Pose Similarity, Transfer Learning

1 INTRODUCTION

Jobs which attribute extended working hours to employees sitting in office desks, have continued to rise for the past decades. Over the past 50 years office jobs have experienced a sharp increase of 80% [2] while back-related issues, specifically low back and neck pains, have emerged globally [3–5] including in the workplace where 30.4% of the employees in participation reported lower back pain during their office jobs [6]. While this might seem like a trivial issue for our well being, it is an issue that is relatively new and its effects on the body occur in the later years of someone's life. Especially for younger individuals, back-related issues do not affect them as much which often causes this issue to be overlooked. Back-related issues can cause fatigue [7] which leads to tiredness and distraction that as a consequence decrease their productivity [8]. To mitigate the problem, companies have begun to provide ergonomic chairs, height-adjustable desks and in-office physicians to their employees. However this does not fully solve the problem, in fact the first two solutions are hard to prove their validity; that even with the most ergonomic chair and desk, the employee will sit properly and not lean or get comfortable in a strained posture.

To prevent the aforementioned, the use of computer vision based on machine learning models to detect when a person is not sitting straight could be an additional solution that can work together with the existing solutions to ensure employees' health. This use case does not have to be limited to an employee with an office desk, albeit the best example, but also for students who study for long periods and any person who wants to work at their home desk. This paper will analyse the performance of 3 models, best suited for human posture image recognition:

(1) **CNN-N:** 3-layered Convolutional Neural Network
(2) **CNN-TL**: The CNN-N model using the MoveNet as a pre-trained model (Transfer Learning)
(3) **LSTM-TL**: A Long Short Term Memory model using the MoveNet as a pre-trained model (Transfer Learning)

Transfer learning, or mentioning the use of a pre-trained model, implies the use of *prior knowledge* in this study which is when a model is using an already pre-trained model to tackle a new problem instead of creating everything from the start. Furthermore, in this paper the term performance, usually in regards to the models, refers to how accurately the model classifies the three postures.

2 RELATED WORK

Due to the existence of Posenet [9], MobileNet [10] and MediaPipe [11] amongst many more, it has become easier to classify body movements from media such as pictures and video. Regarding human posture detection, a study on classifying exercises for shoulder pain and lower back pain was done by comparing media of people performing sets of exercises. The models used for the classification were a CNN and a support vector machine (SVM) as a baseline model [12]. Similarly to this paper, several determinants were used to check the robustness of these models. In the Arrowsmith et al paper they used two different camera angles, coordinate transformations (to recalculate coordinates depending on the position of the bodies to the camera) and the number of keypoints (body parts) used in training. The investigation of these determinants provided a better view of how these variables affected the models. Other related work on human pose detection focused on classes such as sitting, standing, walking, leaning and other similar actions [13–15] instead of human *posture*.

While CNN models are ideal for human pose detection, further research has been done on using Long Short Term Memory (LSTM) models for human poses that are in motion and thus consecutive images can be trained on them [16–18].

However, given these previous works and many more done with a plethora of models; most of them tend to use pre-existing datasets which can be ideal when classifying poses such as sitting, standing and walking but their features can be irrelevant when focusing on specifically sitting postures. To ensure high accuracies, pose similarity equations can aid models with calculating how similar two poses are. Metrics such as the Cosine Pose Similarity [20] and the Object Keypoint Similarity [19] are common standards for this as well as for error diagnosis.

TScIT 39, July 7, 2023, Enschede, The Netherlands

3 RESEARCH QUESTIONS & OBJECTIVES

The following research questions will be answered in this paper:

(1) RQ1: Does prior knowledge affect the model performance for classifying body postures?
(2) RQ2: What physical & environmental determinants can affect the accuracy of these models?
(3) RQ3: Can the model be improved/modified to further improve posture classification?

By answering them, the following objectives can be achieved:

(1) OB1: Provide analysis on the models of the study
(2) OB2: Provide useful datasets for future research

4 METHODOLOGY

A properly organised dataset is needed when comparing multiple models while also addressing RQ2 of investigating physical & environment determinants while a person maintains their posture. People's different body structure, the colour of clothing of the participants as well as the location, which is different for everyone that could use such a model, where the three determinants chosen:

- People: How well does the model classify the posture of different people?
- Location: How well does the model classify the posture in a different environment?
- Clothing: How well does the model classify posture with different types of clothing?

The three determinants are chosen because of their large potential effect on the models. Each of these determinants greatly change the images, such as the different visual aspects of a person: height and size or the different location which can greatly affect the background: wall with a colorful painting in the back, glass panels of the office building, background with many objects or finally the clothing of the person: how baggy or tight it is and its colors. All three determinants either affect the image significantly through color or the area of the image that they cover and for these two reasons their influence to the models' performance will be studied in this paper. Other possible determinants include lighting, type of camera or device used, camera-to-person angle, person-to-camera angle, and image quality to name a few.

4.1 Data Collection

The purpose of gathering data is to have more valid datasets within the scope of identifying sitting postures. The participants included university students and employees with an age range of 20 to 31 years old. The sample (n=14) was predominantly European Caucasian with 2 participants from North Africa. The participants where asked to sit in three different sitting positions categorized as leaning to the left, leaning to the right and sitting straight. For each participant, their postures were video recorded from the same camera on the researcher's laptop, to maintain the same resolution. Video recording enables a faster way of capturing relevant images as well as maximizing the quantity, compared to taking multiple individual photos. Participants were recorded in various locations at the University of Twente, multiple houses in Enschede and one office in Amsterdam. Participants also wore various colors and types of clothing (sweaters and t-shirts) in order to generalize the training for the models. The procedure of the recordings involved a quick brief with instructions and the purpose of the study, followed by the participants sitting down so that the recording can begin. They were asked to first sit straight, then lean their upper body to the left and then lean to the right. They were asked to repeat these three postures once more. If needed, the researcher showed them how to move. The dataset consists of 2751 images in total with three classes: left, right and straight.

4.2 Data Processing

4.2.1 CNN models. For the CNN models (CNN-N and CNN-TL), data organization was automated to an extent: a script was used to capture an image every 5 frames of the recording. Manually these images were either removed if a pose was very close to being a straight pose while at the same time the person was in-motion to leaning to one direction. These moments in images would confuse the model, even humans, and consequently were not included.

The images were then downsized from a resolution of 780x1280 to smaller resolutions: 256x256 (CNN-N) and 224x224 (CNN-TL). They were also stretched, via automated cropping, in order for the leaning to be more noticeable across the x-axis. Lastly, body segmentation was used to remove the background from the images.

4.2.2 LSTM model. For the LSTM model, due to its temporal nature, a different method of data processing was used. By capturing each image every 5 frames, the chronological sorting was crucial such that batches of 10 images could be inputted into a time distributed layer prior to the LSTM layer. Due to this process, fewer images out of the complete dataset were used because consecutive images which had no significant changes in the participants' bodies would not be inserted in the batch and thus in total only 1536 images where used. Body segmentation was also used.

4.3 Testing

By acquiring the dataset and processing it accordingly, the data can be fed into the models. To properly test the results two procedures were used: testing the validation accuracy and loss, and using k-fold cross validation. The former calculates the loss, which is the sum of errors in the model and the accuracy, how well the model correctly classifies the data, on the validation set. The validation set is part of the dataset that is used to test how well the model performs. For the k-fold cross validation, the dataset is split into k groups and for each group, the rest of the k-1 groups are used as the training data while the k-th group is used as the validation data (data to be tested on). An average of the accuracy of all the k-trials is made to evaluate how well the model performs. This is to ensure that no matter how the dataset is split, the average accuracy will be a valid metric of the models performance.

5 MODELS

5.1 Model Architecture

The main focus of this research is to investigate whether including a pre-trained model would improve the model accuracy over the normal model. Therefore it is crucial to investigate how well the CNN via transfer learning (CNN-TL) and the LSTM via transfer learning (LSTM-TL) will be performing against the normal CNN model (CNN-N). It was not possible to compare a normal LSTM model with the LSTM-TL, to investigate the effect of prior knowledge, since by default when using an LSTM model for image detection and not for data forecasting - at least one CNN layer is needed. This does not make it a standalone LSTM model anymore, yet a CNN-RNN model (which used a GRU layer instead of a LSTM layer) is discussed and compared to the other three models in Section 7.

5.1.1 CNN model. The CNN-N follows a rather straightforward layer configuration of filters of increasing power of 2, followed by 2D Max Pooling and a constant 0.75 dropout rate after every convolutional layer. Given the complexity of the images, there was no need for the model to go deeper and include any additional convolutional layers. At the same time, the highest filter was 64, which was enough to yield the desired results. A softmax function was used in the final Dense layer.

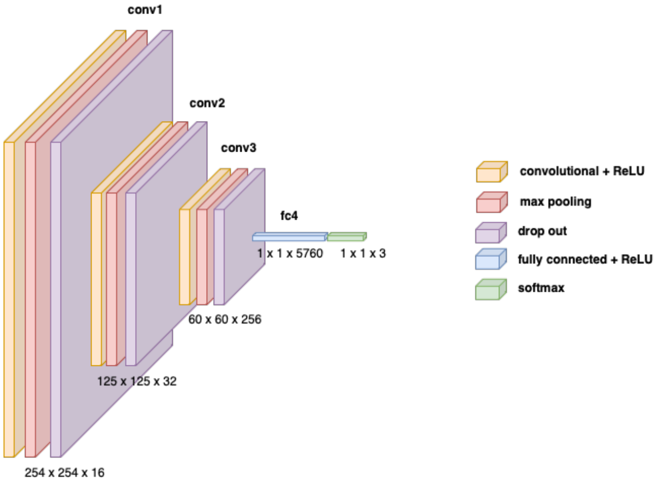


Fig. 1. Layer Configuration of the CNN-N

5.1.2 CNN via Transfer Learning. This model uses the TensorFlow SinglePose MoveNet (Lightning) model as the base model which provides very useful skeletal features of the body that can then be used, with a Flatten layer followed by two Dense layers.

5.1.3 LSTM via Transfer Learning. The idea for such a concept rose from the case that the postures to be classified are in-motion actions such as a person leaning to the left while sitting. Therefore the idea to combine the spatial abilities of a CNN such as the one used in MoveNet model with the temporal abilities of an LSTM were explored. By being in-motion, the LSTM model can remember the movement of a person who is moving to one side using its short term memory which can learn from previous images that are chronologically sorted or through video input.

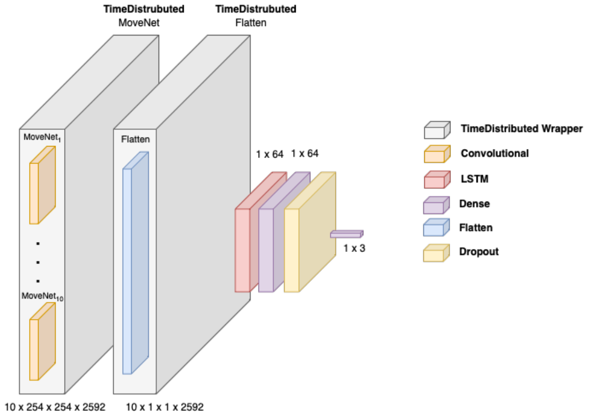
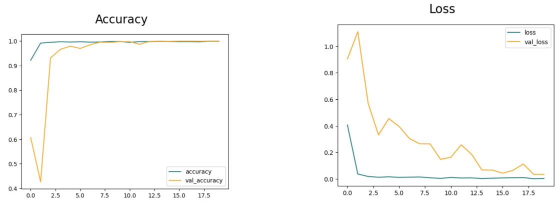


Fig. 2. Layer Configuration of LSTM-TL

5.2 Testing Procedures & Results

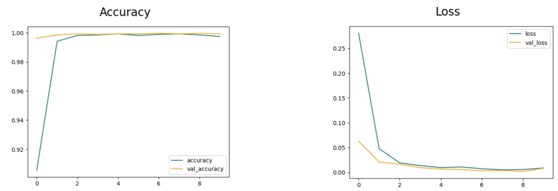
For training loss, Keras does a running average over the batches. For validation loss, a conventional average over all the batches in validation data is performed. The training accuracy is the average of the accuracy values for each batch of training data during training. Furthermore, K-Fold validation was performed to show consistency across the entire dataset.

5.2.1 Validation Accuracy and Loss. The accuracy of CNN-N and CNN-TL averaged 96.03% and 99.3% over 20 epochs respectively while the LSTM-TL performed slightly worse with an accuracy average of 93.7% over 50 epochs. The losses were 0.012, 0.0016 and 6.1 on average for CNN-N, CNN-TL and LSTM-TL respectively. The accuracies and losses are plotted on Figures 3-5 where the yellow line is the validation accuracy or loss and the green line the training accuracy or loss. For the CNN-N (Figure 3) the validation loss is higher than the training loss however this is not a sign of overfitting as the validation loss is decreasing over the epochs. For the LSTM-TL (Figure 5) the loss is fluctuating due to insufficient data as suggested through the spread of accuracies during the K-fold testing in Table 1.



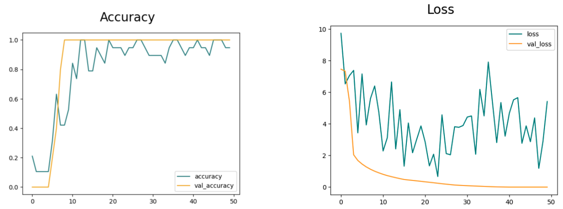
(a) Accuracy (b) Loss

Fig. 3. CNN-N Accuracy and Loss



(a) Accuracy (b) Loss

Fig. 4. CNN-TL Accuracy and Loss



(a) Accuracy (b) Loss

Fig. 5. LSTM-TL Accuracy and Loss

5.2.2 *K-Fold Cross Validation.* The three models were validated over 20 epochs excluding the LSTM-TL which needed 50 epochs.

Results (k=5)			
Folds	CNN-N	CNN-TL	LSTM-TL
1	0.9946	0.9964	0.7272
2	1.0000	0.9982	0.6999
3	1.0000	0.9964	0.8999
4	0.9945	0.9982	0.8000
5	0.9982	0.9964	0.8999
μ	0.997	0.997	0.805
σ	0.00246	0.00088	0.08383

Table 1. K-fold Validation of the three models

K-Fold testing in Table 1 shows that the CNN-N and the CNN-TL perform similarly with low spread of averages while the LSTM-TL model maintains a worse average, with more unpredictable results on the folds. This unpredictability can be due to the smaller subset of the original dataset that was used for the LSTM-TL model, amongst other reasons which are discussed more in-depth in Section 9.

6 DETERMINANTS OF MODEL PERFORMANCE

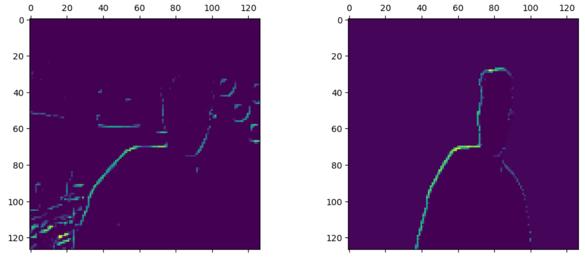
Environmental determinants can be considered as attributes of the background behind the person as well as lighting that can affect model performance while physical determinants in this study can be described as physical attributes of the person that can affect the performance. Various metrics were used to calculate noise in images and color differences. The results of such metrics are discussed in Section 6.3.

6.1 Environmental Determinants

6.1.1 *Background (Location) as a Determinant.* In this section the location, which can be seen as the background in an image, can also affect the accuracy of the models.

Taking a deeper dive into individual layers of the CNN-N model, on the 1st convolutional layer (filter=16), after the max pooling process as seen in Figure 6 there are a few refined edges from the background on objects such as a mirror, vinyl records on the wall and shoe drawers. These edges can negatively affect the results on the later layers of the model.

However, if the background is removed then these edges disappear and the edge from the participant's arm stretches over their shoulder until their head is more visible. In fact in Figure 6a, this edge was going over the person's desk chair instead of their shoulder because the chair edge was more noticeable to the model while after the body segmentation the shoulder is smaller because the edge is not passing over the chair anymore.



(a) before body segmentation (b) after body segmentation

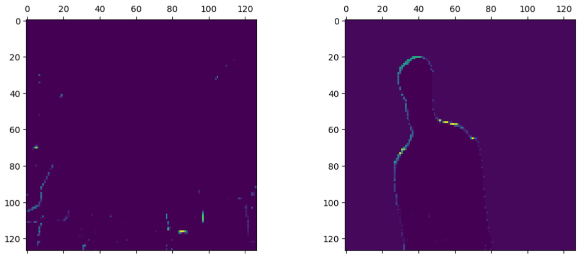
Fig. 6. A heatmap of the 1st Conv2d layer (+ max pooling) showing a case of a noisy background

To measure the level of effect that backgrounds had to the accuracy of the models, the Signal-to-Noise Ratio (SNR) which compares the image content to visual "noise" in order to measure the noise of the given image. A higher SNR means a stronger signal of image content and thus better quality with lower noise.

6.1.2 *Quality of Light as a Determinant.* The amount of lighting has an important role in the accuracy and consistency of the model results. Participants recorded in an environment of lower light resulted in more grainy images that resulted in generally worse results. Sun glares also seemed to cause a similar issue where participants had a lot of light, however this case yielded better results than the low light case.

In Figure 7, the background sunlight is overexposing the image creating a lack of edges for the model to work with. Again, when the background is removed the edges of the participant are more visible.

The quality of light depends on whether an image is overexposed, underexposed or if there is uneven light. When there is too little light or sub-par quality of light, the signal of the SNR is weak which in return lowers the SNR value indicating that there is a substantial amount of noise in the image. Conversely, a lot of light leads to a higher signal because more light is captured and less noise. While



(a) before body segmentation (b) after body segmentation

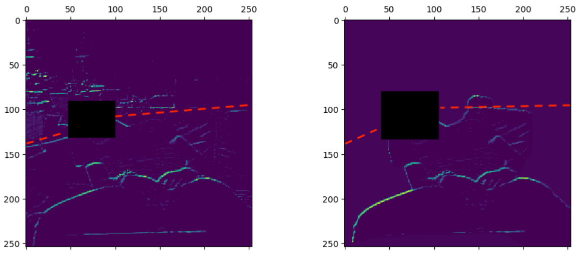
Fig. 7. A heatmap of the 1st Conv2d layer (+ max pooling) showing a case of bad lighting

this might sound optimal, a lot of information of the image is lost because it is covered by the excess light. Due to this similarity that low lighting has with the noise of an image, the SNR results in Table 2 can be used as a metric of how sub-par lighting affects the models as well.

6.2 Physical Determinants

6.2.1 Image Color as a Determinant. Images where the person's clothing matches the background e.g. the couch or wall, can have a lower accuracy than the images where this was not the case. Due to this, even the body segmentation process had occasional issues detecting the body with the background. These images were later manually edited to remove the leftover background yet the model improvement was insignificant.

In Figure 8, the red dotted lines on each image denote the axis of a black couch. The participant in the image is also wearing a black t-shirt and as a result the edges are not as refined because the model cannot differentiate the silhouette of the participant from the couch. After the body segmentation a part of the couch is still visible to the right of the participant's head. This is an example of color matching affecting the body segmentation process.



(a) before body segmentation (b) after body segmentation

Fig. 8. A heatmap of the 1st Conv2d layer (+ max pooling) showing a case of color matching, the color of the participant's t-shirt matches the color of the couch. The black box is used to cover the participant's identity.

To measure the effect of matching colors with the background and the participant, a sophisticated procedure had to be used due to the complexity of comparing colors. There are two school of thoughts when it comes to color difference: using the algebraic distance of a color space such as RGB, LAB, Y'UV and many morem, which is how computers understand the difference between two colors or to use more complex equations that take in account the lightness, hue and other color factors when determining the color difference. The latter more complex methods try to imitate how the human eye perceives color differences, as opposed to the "eye" of the computer. Even though we are dealing with a CNN model which learns through calculation, a latter method was chosen to analyze the variety of backgrounds more carefully. The CIEDE2000 formula [21], takes into account the lightness, colorfulness and hue of the colors when calculating their difference in the LAB color space. The average of the top 2 dominant colors were taken from the body segmented image (only the participant) and the background image were used as the two colors which were plugged in the CIEDE2000 formula.

6.2.2 Clothing as a Determinant. Another physical determinant to consider is the clothing of a person. Since the color matching is already discussed; the type of clothing can be also investigated, specifically the amount of clothing. All participants were split into two groups: the participants (n=8) with sweaters or clothes that covered their arms (long sleeve group: LS) and the participants (n=6) with short sleeve clothing such as t-shirts and tank tops (short sleeve group: SS).

On average, the SS group accurately classified the images better by 18% on the left class, 22% on the right class and only 1% on the straight class. For the first two results, the better performance can be attributed to more skin showing on the arms and sometimes on the shoulders as was the case with tank tops and as such it was easier for the model to outline the edges than it was with a baggy sweater. The similar performance on the straight class has to do with the low variance that the straight posture has. For the left and right postures, participants could be leaning in different ways and to different extents while for straight postures, all the participants have a very similar straight posture and due to that the improvement was insignificant on that class.

Going back to Figure 8, if the participant was to wear a black long sleeve sweater the horizontal edges on their arm, denoting the end of the t-shirt and the beginning of the skin, would not be visible. These edges can help the model understand the skeletal position of the participant so this is a case where less clothing helps the model in finding the important edges.

6.2.3 People as a Determinant. Lastly, to address the last determinant of how different people can affect the model; the participants size, height, age, gender and skin color did not affect the model in detecting the edges and subsequently the posture. This can be due to the sophisticated methods of the CNN architecture which can learn very well to detect people but can also be due to the lack of diversity in the sample as discussed more in-depth in Section 10.

6.3 Determinant Metrics

The SNR and the CIEDE2000 were calculated on 4 pre-selected backgrounds. Images taken of participants at a house with just a white wall in the background (BG-1), a house with a typical background of a kitchen (BG-2), backgrounds that were mainly covered by glass (BG-3) and a room with many objects in the background (BG-4). In Table 2, the model confidence is measured as the average of the *straight* posture probability, which is different from the model accuracy which would be defined as how many times the straight posture is correctly classified. The metrics were calculated only with the CNN-N model for simplicity. Per background, 100 images of straight postures of two participants were selected for evaluation.

Determinant metrics			
Background	SNR	CIEDE2000	Model Confidence
BG-1 (white)	-13.1, -15.4	113, 111	0.72
BG-2 (kitchen)	-13.3, -13.9	111, 111	0.47
BG-3 (glass)	-14.1, -15.0	115, 114	0.24
BG-4 (noisy)	-14.4, -15.1	104, 113	0.12

Table 2. Results of the SNR and the CIEDE2000 compared to the model confidence on 4 different backgrounds

A few conclusions can be made from these results. The lower the negative SNR value, the more noisy the image was, which lowered the model confidence. The BG-4, which was the background of a room with multiple objects, which can be vaguely seen in Fig. 6a, had the lowest SNR value as expected but also the lowest model confidence. A model confidence of 0.12 indicates that the average of the straight posture classification probabilities was approximately 0.12 while for the left and right it would have been most likely even lower. This means it still classified the straight posture correctly yet with a very low confidence, due to the high noise levels.

For the CIEDE2000 results, a higher value indicates high color similarity between the participant and the background. Most values are surprisingly similar to each other and concrete conclusions cannot be made. While raw images cannot be shown due to the signed consent of the participants, in BG-1 the two participants wore light colored sweaters slightly matching the background. The model confidence for BG-1 is rather high for such a color similarity which should normally confuse the model. BG-3, which was a glass background, has the highest color similarity mostly because the incoming sunlight entering the glass smoothed all the colors to a slightly dimmer color palette across the whole image. At the same time, the incoming light underexposed the foreground parts of the image creating more noise as Table 2 shows.

Overall, it seems that the color difference/similarity does not seem to affect the confidence of the model as much as noise does. Therefore the more grainy, lower quality and underexposed an image is the more it can confuse the model to a greater significant effect than color similarity. The background which was mostly a white wall had the highest model confidence, despite not having the least noise or the lowest color similarity, they were still lower than some other more complicated backgrounds.

7 FURTHER RESEARCH ON LSTM/RNN MODELS

During the study of the aforementioned models, another model was being developed on the side yet was not mentioned together with the rest to keep the model comparisons simple when discussing whether prior knowledge is important. However, when it comes to the overall research of human posture detection, this hybrid model can be a possible alternative. It extends the idea of the LSTM-TL model where a pre-trained CNN model is used to teach the spatial features to the LSTM model which focuses on the temporal features. The hybrid model was implemented by using the same layer configuration of the CNN-N model as input to an RNN model and as a consequence combining the layers of both models.

Note that this model's layer configuration is the same as the one from the LSTM-TL in Figure 2 but the LSTM layer is now replaced by a Gated Recurrent Unit (GRU) layer and instead of the pre-trained CNN model inside the Time Distributed layer, it is the CNN-N model. LSTM models are a type of Recurring Neural Network (RNN) models and in this case using a GRU layer with the same amount of 64 neurons yielded better results than the LSTM layer with the same number of neurons. Ideally both the GRU and LSTM layer work in this case, their only difference being that the LSTM layer has 3 gates compared to 2 because the GRU layer does not have a memory gate. This difference makes the GRU layers more efficient but, in the case of the small datasets used in this study, this efficiency gap is not noticeable.

Results (k=5)				
Folds	CNN-N	CNN-TL	LSTM-TL	**CNN-RNN**
1	0.9946	0.9964	0.7272	0.7750
2	1.0000	0.9982	0.6999	1.0000
3	1.0000	0.9964	0.8999	0.9512
4	0.9945	0.9982	0.8000	0.8293
5	0.9982	0.9964	0.8999	0.9756
μ	0.997	0.997	0.805	0.906
σ	0.00246	0.00088	0.08383	0.08805

Table 3. K-fold Validation of the models with the CNN-RNN

Table 3 shows the results of all the models together. The CNN-RNN model has a better overall accuracy than the LSTM-TL but its results are also more unpredictable on every fold.

8 MODEL IMPROVEMENTS

8.1 Optimizers

While these optimization methods were already included in the three models and thus in their results, it is worth mentioning that high dropout rates were tested via trial and error as well as learning rates and layer weight kernel regularizers. For the latter two, the CNN-N and CNN-TL models needed learning rates of 0.0001 and 0.001 on each model respectively with the CNN-N model having two layer weight regularizers of 0.0001 on the 2nd and 3rd convolutional layers. The LSTM-TL had a learning rate of 0.0001 with no kernel regularizers.

8.2 Pose Similarity Metric

8.2.1 Cosine Pose Similarity. During the pilot phase of the testing several of the images used were taken using a smartphone. Even when those images were resized, several of them would get misclassified. This was the main motive in looking into the cosine pose similarity equation, which shows how similar two vectors are by dividing their dot product with the product of their magnitude:

$$C(u,v) = \frac{\vec{u} \cdot \vec{v}}{||\vec{u}||||\vec{v}||}$$

On its own this formula outputs a matrix of [0,1] values where a vector u is more similar to v if it is closer to 1. By using the MediaPipe Pose Estimation model [22] to detect all the body parts of a person, this equation when mathematically fused with the classification probabilities should help solve the model mis-classification issue. In the case of sitting posture the important vectors for the pose similarity would be the left and right shoulder coordinates and the left and right hip coordinates; these can be referred to as *focal coordinates*.

8.2.2 Augmented Cosine Pose Similarity Metric (aCPSM). However the cosine pose similarity formula on its own does not give results that can fix the mis-classifications, instead its results coincide with the mis-classified softmax probabilities from the models. Therefore, the cosine pose similarity equation needs to be modified. One way to start would be to compare the given pose with a ground truth pose, a pose where its classification is certain, of each of the classifications.

Although, how can we acquire the ground truth poses and be certain they are correctly classified? One way is to randomly select a batch of images from each class in the training dataset and calculate the average focal coordinates of each pose in order to acquire the ground-truth poses. This ensures generality, since it involves different people in different locations. Let the ground-truth poses be defined as:

$$\mathcal{T}_\gamma = \frac{\sum_{i=1}^{B_\gamma} \begin{pmatrix} ls_x^{(i)} & rs_x^{(i)} \\ lh_x^{(i)} & rh_x^{(i)} \end{pmatrix}}{B_\gamma}$$

Where γ of $\mathcal{T}$ is the classification label, B_γ is the batch size of γ, and ls, rs, lh and rh are the focal coordinates: left/right shoulders and left/right hips respectively. The $\mathcal{T}_\gamma$ equation adds the x values of each focal coordinate of B_γ number of images and then divides it by B_γ to get the average of each focal coordinate in matrix form. Now each ground truth pose, in this case: $\mathcal{T}_{left}, \mathcal{T}_{right}, \mathcal{T}_{straight}$ can be compared to the given (unseen) pose, P, in order to calculate the cosine pose similarities:

$$C^{(\gamma)} = C(\mathcal{T}_\gamma, P)$$

Using the acquired ground truth cosine pose similarities, which consists of a matrix of 4 probabilities (the number of the focal coordinates), the probabilities of the shoulders are subtracted by the give pose shoulder x-values (percentage). This is the augmented cosine pose similarity metric (aCSPM) defined by δ_γ:

$$\delta_\gamma = \frac{\sum_{c=0}^{C_f} |C_{c,0}^{(\gamma)} - P_{c,0}|}{C_f}$$

Where C_f is the number of focal coordinates. By using MediaPipe these x-values are measured as a percentage of the screen's width instead of the number of pixels. It is only in the human posture scenario of this study, that the y-values where omitted from the δ_γ equation (shown by iterating only over c in $C_{c,0}$ and $P_{c,0}$) due to the nature of the horizontal movement of the upper body that needs to be classified. The δ_γ equation can be generalized and used with any model focused on body movements that needs further improvements in performance. For example, in the case of detecting people jumping, the focal coordinates would be the knees and the feet. In this situation the δ_γ would be subtracting the y-values of the focal coordinates rather than the x-values and thus the absolute value would be $|C_{0,c}^{(\gamma)} - P_{0,c}|$. In the case of needing to use x-values and y-values, their product can be used: $|C_{c,0}^{(\gamma)} - P_{c,0}||C_{0,c}^{(\gamma)} - P_{0,c}|$.

Finally we can combine the δ_γ with the classification probabilities and a model-dependence bias:

$$\eta_\gamma = (1 - \delta_\gamma)^{f(\gamma)+b}$$

$$\hat{\eta} = \begin{pmatrix} \eta_{\gamma_0} \\ \vdots \\ \eta_{\gamma_n} \end{pmatrix}$$

Where $f(\gamma)$ is the classification probability of γ class and b is the model-dependence bias which serves the purpose of dictating the sensitivity of $f(\gamma)$ to the final classification probabilities: $\hat{\eta}$. By setting $b = 0.5$, the $f(\gamma)$ can be sensitive enough to the δ_γ metric so that when needed, the δ_γ has enough influence to the final η_γ accuracy.

8.3 Results Based on the Augmented Cosine Pose Similarity Metric

The dataset used for this testing procedure varies greatly from the dataset used up to now. The images used were taken from a smartphone during the pilot phase. The angles of the camera, due to how smartphones stand upright, varied greatly compared to the images captured by the laptop. The images can, out of coincidence, be described as hard to predict for the models since the smartphone camera performs significantly worse than the laptop camera in dim light, participants' postures were sometimes exaggerated and the camera resolution was lower. Lastly, the dataset is much smaller consisting of an average of 10-20 images per label.

Due to the different method of data collection, mainly that the smartphone photos were taken individually rather than through video recording, the LSTM-TL and the CNN-RNN were omitted from this testing procedure because there were no consecutive images.

Results		
Class	CNN-N	CNN-TL
Left	0.8095	0.7778
Right	0.9017	0.9215
Straight	0.2051	0.6154
μ	0.639	0.772

Table 4. Accuracy results without the aCPSM

Results with aCPSM		
Class	CNN-N	CNN-TL
Left	0.8571	0.8714
Right	0.7843	0.9020
Straight	0.8462	0.8205
μ	0.829	0.865

Table 5. Accuracy results with aCPSM

The results show how when the CNN-N and CNN-TL, which are trained with the original dataset, are given a set of unseen images from the smaller dataset with different resolution and quality, they perform worse than their accuracies shown in Table 1. However by using the aCPSM metric, these accuracies are greatly improved.

9 DISCUSSION

As shown from the prior testing procedures throughout the study, the CNN-TL and the CNN-N models perform the best via the K-Fold Cross Validation and the Validation Loss/Validation Accuracy testing. These results would imply that having prior knowledge in a model does not significantly improve or worsen the model's performance in the context of human posture classification. However the aCPSM results on Table 3 that were derived from the much smaller and unpredictable dataset, showed that the CNN-TL model performs better than the CNN-N by more than 10% (77.2% vs 63.9%). Due to this a subsequent test was done by retrieving 543 images from the original dataset, and testing this subset on 102 unseen images. The CNN-TL outperformed the CNN-N by almost 10% this time (96% vs 88.3%). With these results it can be concluded that when the dataset is extremely small, prior knowledge can improve the performance. However, after a certain dataset size, which said size depends on the type of data and complexity of the classification, the prior knowledge does not improve the performance which is the case in Table 1. Or perhaps the improvement is extremely small. Regardless, the choice of whether to use a model with prior knowledge or not could be an important decision when training models with an extremely large dataset. The dataset of this study has less than 3000 images, but if it were to be >50000, the model without the prior knowledge could be significantly faster while yielding similar accuracies to the model with prior knowledge.

On the other hand, comparing the LSTM-TL, which has prior knowledge, with the CNN-RNN it can be seen that both need a larger dataset due to the nature of the data processing. Their input can be seen as a batch of images instead of one image at a time, and as a result by taking 10 images at a time for each batch the model has fewer inputs to work with compared to the CNN models. The batch size was also increased to 15 to verify whether more consecutive images provide the models with more data to work with, yet the results were either similar to a batch size of 10 or worse. By inspecting the individual K-fold results of the LSTM-TL and the CNN-RNN models, the latter has a larger spread of averages making it more unpredictable. It could be argued that this is not the case with the LSTM-TL because it also has the prior knowledge of the MoveNet and consequently makes the results more consistent. While there is no proof derived from this study: it could be that with a larger dataset the CNN-RNN could still outperform the LSTM-TL model because the MoveNet does not organize its data in batches as the RNN models require and almost zero temporal features can be extracted by remembering consecutive images.

Lastly, on topic with choosing the right pre-trained model, the LSTM-TL was also trained on the InceptionV3 pre-trained model yielding a k-fold (k=5) accuracy of 71.8% which is approximately 8% lower than with the MoveNet model (80.5%). This result further shows how important choosing the right pre-trained model as well as when to use them and when not depending on the size of the dataset.

10 LIMITATIONS OF THE STUDY

The most important limitation to this study is the dataset itself. While it attempts to further improve existing studies on human posture by having a more realistic dataset, the data itself is not enough to be generalized into detecting individuals' postures across the world, due to the limited environments, lack of ethnicities and age groups of the participants who were recorded. While the age range was reported as 20-31 years old in the Methodology section, most of the sample had the range of 20-23 with only 2 outliers to that range. Furthermore, the physique of the participants on average was very fit, which is to be expected with a young predominant age range of 20-23. People with disabilities such as body deformations, individuals on wheelchairs (lower than a desk chair) and individuals with missing limbs were not part of the study. For these reasons the People physical determinant was studied less in this paper and its results of not affecting the model does not extend to all individuals but rather to the dataset that the model was trained on. This should be a dataset that continues to grow, as office jobs and employees increase, to keep up with the spatial and color diversities in the images to be detected.

11 CONCLUSION

Three neural network models, of which two used pre-trained base models, were tested on a dataset of three different postures. Various testing procedures were conducted mainly Validation Loss/Validation Accuracy testing, K-Fold Cross Validation and the Unseen Validation Set testing, where it was concluded that normal CNN model (CNN-N) performs similarly to the CNN with prior knowledge (CNN-TL) when the dataset is large enough size which in the context of this study would be approximately 2400-2800 images. Through further testing, it was shown that prior knowledge can improve performance by 10% when the dataset is extremely small, which for this study was approximately 500 images. For extremely large datasets, it could be wiser to sacrifice the prior knowledge for efficiency and speed. Finally, it is vital to use body segmentation since environmental determinants such as the background can significantly alter the models' performance through noise. To a lesser but still significant extent, the clothing of a person should be considered since having less clothing can help the model understand the skeletal position better and lastly the type of people, given the dataset, did not affect the models to any significant extent. Despite that, it was shown that the augmented cosine pose similarity can aid the CNN-N and CNN-TL models to minimize the negative effects of the physical, environmental and camera-related determinants.

REFERENCES

[1] Kim, D., Cho, M., Park, Y., Yang, Y. (2015). Effect of an exercise program for posture correction on musculoskeletal pain. Journal of Physical Therapy Science, 27(6), 1791–1794. https:// doi.org/10.1589/jpts.27.1791

[2] Church, T. S., Thomas, D. M., Tudor-Locke, C., Katzmarzyk, P. T., Earnest, C. P., Rodarte, R. Q., Martin, C. K., Blair, S. N., Bouchard, C. (2011). Trends over 5 Decades in U.S. Occupation-Related Physical Activity and Their Associations with Obesity. PLOS ONE, 6(5), e19657. https://doi.org/10.1371/journal.pone.0019657

[3] Clark, S., Horton, R. (2018). Low back pain: a major global challenge. The Lancet, 391(10137), 2302. https://doi.org/10.1016/s0140-6736(18)30725-6

[4] Driscoll T, Jacklyn G, Orchard J, et al The global burden of occupationally related low back pain: estimates from the Global Burden of Disease 2010 study Annals of the Rheumatic Diseases 2014;73:975-981.

[5] Hurwitz, E.L., Randhawa, K., Yu, H. et al. The Global Spine Care Initiative: a summary of the global burden of low back and neck pain studies. Eur Spine J 27 (Suppl 6), 796–801 (2018). https://doi.org/10.1007/s00586-017-5432-9

[6] Zafar F, Qasim Y F, Farooq M, et al. (August 22, 2018) The Frequency of Different Risk Factors for Lower Back Pain in a Tertiary Care Hospital. Cureus 10(8): e3183. DOI 10.7759/ cureus.3183

[7] Ghamkhar, L., Kahlaee, A. H. (2019). The effect of trunk muscle fatigue on postural control of upright stance: A systematic review. Gait Posture, 72, 167–174. https://doi.org/10.1016/ j.gaitpost.2019.06.010

[8] Ricci, Judith A. ScD, MS; Chee, Elsbeth ScD; Lorandeau, Amy L. MA; Berger, Jan MD. Fatigue in the U.S. Workforce: Prevalence and Implications for Lost Productive Work Time. Journal of Occupational and Environmental Medicine 49(1):p 1-10, January 2007. | DOI: 10.1097/01.jom.0000249782.60321.2a

[9] Badrinarayanan, V., Grimes, M., Cipolla, R. (2015). PoseNet: A Convolutional Network for Real- Time 6-DOF Camera Relocalization. arXiv (Cornell University). https://doi.org/10.48550/ arxiv.1505.07427

[10] Bajpai, R. and Joshi, D., "MobileNet: A Deep Neural Network for Joint Profile Prediction Across Variable Walking Speeds and Slopes," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-11, 2021, Art no. 2508511, doi: 10.1109/TIM.2021.3073720.

[11] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C., Yong, M. G., Lee, J., Chang, W., Hua, W., Georg, M., Grundmann, M. (2019). MediaPipe: A Framework for Building Perception Pipelines. arXiv (Cornell University). https://doi.org/ 10.48550/arxiv.1906.08172

[12] Arrowsmith, C., Burns, D. J., Mak, T., Hardisty, M., Whyne, C. M. (2022). Physiotherapy Exercise Classification with Single-Camera Pose Detection and Machine Learning. Sensors, 23(1), 363. https://doi.org/10.3390/s23010363

[13] Ogundokun, R. O., Maskeliūnas, R., Misra, S., Damasevicius, R. (2022). A Novel Deep Transfer Learning Approach Based on Depth-Wise Separable CNN for Human Posture Detection. Information, 13(11), 520. MDPI AG. Retrieved from http://dx.doi.org/10.3390/info13110520

[14] Haque, S., Rabby, A.S.A., Laboni, M.A., Neehal, N., Hossain, S.A. (2019). ExNET: Deep Neural Network for Exercise Pose Detection. In: Santosh, K., Hegadi, R. (eds) Recent Trends in Image Processing and Pattern Recognition. RTIP2R 2018. Communications in Computer and Information Science, vol 1035. Springer, Singapore. https://doi.org/10.1007/978-981-13-9181-1_17

[15] Liaqat, S., Dashtipour, K., Arshad, K., Assaleh, K. and Ramzan, N. (2021) A hybrid posture detection framework: Integrating machine learning and deep neural networks. IEEE Sensors Journal, (doi: 10.1109/JSEN.2021.3055898)

[16] Lee K., Kim W. and Lee S., "From Human Pose Similarity Metric to 3D Human Pose Estimator: Temporal Propagating LSTM Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 2, pp. 1781-1797, 1 Feb. 2023, doi: 10.1109/TPAMI.2022.3164344.

[17] K. Fragkiadaki, S. Levine, P. Felsen, J. Malik; Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 4346-4354

[18] Nie B. X. , Wei P. and Zhu S. -C., "Monocular 3D Human Pose Estimation by Predicting Depth on Joints," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 3467-3475, doi: 10.1109/ICCV.2017.373.

[19] Ronchi, M., Perona, P. (2017). Benchmarking and Error Diagnosis in Multi-Instance Pose Estimation.

[20] D. Wang, H. Lu and C. Bo, "Visual Tracking via Weighted Local Cosine Similarity," in IEEE Transactions on Cybernetics, vol. 45, no. 9, pp. 1838-1850, Sept. 2015, doi: 10.1109/TCYB.2014.2360924.

[21] Sharma, G., Wu, W., Dalal, E. N. (2004). The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. Color Research and Application, 30(1), 21–30. https://doi.org/10.1002/col.20070

[22] Pose landmark detection guide. (n.d.). Google for Developers. https://developers.google.com/mediapipe/solutions/vision/pose$_landmarker$