Human/Animal Activity Recognition Data Analysis - Classification and Identification

ARYAN KHAWANI, University of Twente, The Netherlands

With the rise of technology in the modern era, many pet owners and animal caretakers in general need effective and accurate activity detection. Additionally, people with elderly parents or grandparents at home want to know what their family member is doing, and if they are fine. Traditional methods of activity detection with cameras have their drawbacks, however, since cameras can be affected by only what is visible by the naked eye. Therefore, this study performs activity classification using Micro-Doppler Signature (MDS) generated from FMCW radars, comparing the differences between six different classification models for six different activity classes, which resulted in accuracies from 84% to 93% when trained and validated on a dataset of 886 Micro-Doppler Signature spectrogram images with a 20% validation split. The study goes further and proposes an architecture for a Siamese model for the identification of the subjects performing the action with an F1-score of 84% for similar pairs and 81% for dissimilar pairs. The study then concludes by adding how this work can be built upon in the future. This study is one of the first for animal activity classification and identification using MDS images.

Additional Key Words and Phrases: Micro-Doppler Signature, FMCW radar, Intermediate Frequency Signal, image classification, deep learning, activity detection, data analysis, CNN, Bidirectional LSTM, Transfer Learning, Contrastive Learning, Siamese Network

1 INTRODUCTION

As technology in the world develops, the demand for effective, efficient and accurate systems to detect the activity and well-being of humans and animals alike has skyrocketed. There are multiple reasons for this increase in demand. One use is tracking the movement and vital signs of a pet to ensure it is healthy. Tracking how certain people move, especially those of age, to watch over them and detect if they had a random fall, for example, is also important. These systems can also detect suspicious human activity to safeguard the people of the world [15, 32]. It also becomes important to identify the subject who took the action. For example, if you have multiple pets, you would want to know which pet is currently eating. Likewise, you might need to identify the elderly person who fell so you can call them an ambulance remotely. As for suspicious activities, you would want to ensure that only the person responsible suffers the consequences.

Although there are some common methodologies for activity detection of humans and animals for personal and social well-being, they have some drawbacks. For example, in the case of using video cameras for surveillance footage, it could be the case that the objects visible on the screen for analysis are affected by various conditions such as lighting, the position of the camera or even the quality of the video output. When using deep neural networks like in [32], it is quite likely that the computer will struggle to comprehend the figures and activity, given these conditions. Additionally, various wearable devices could be required for the subject to get more in-depth information of vital signs and activity, which may cause them discomfort, physically and mentally.

To address this issue, we propose employing sensors with the purpose to acquire Frequency Modulated Continuous Wave (FMCW) radar [1, 16, 42]. These sensors emit FMCW waves, which can then be analyzed to produce an intermediate frequency (IF), commonly referred to as a beat signal. This beat signal contains crucial physiological data such as heart rate, respiration rate, and movement patterns without requiring direct contact with the subject [24]. This data can be instrumental in assessing the health of a pet or understanding the actions of both humans and animals. However, to utilize this information effectively, it must first be processed into a visual format.

To visualise the activity and movement, the IF signal must be processed to obtain a Micro-Doppler Signature (MDS), in the form of a spectrogram, As the name suggests, the Micro-Doppler Signature would be unique to distinguish from the other signatures generated, with respect to the subject's activity [35].

In this paper, we analyze Micro-Doppler Signature images generated from a dataset containing activity data captured by FMCW radar sensors. The dataset contains six unique activity classes, each containing 72 MDS images. Each activity is also split into two subjects performing the action. For image classifiers, the amount of data present may not be enough, however.

For the provided dataset, the list of activities are as follows:

- Cat Eating
- Cat Jumping
- Cat Walking By
- Cat Walking Towards
- Cat Walking with Human
- Human Walking

The goal of this study is to leverage Deep-Learning (DL) models for for two key tasks: *(i) classification* and *(ii) identification*. Firstly, we seek to classify various activities observed in the dataset, using DL image classifiers with the generated MDS spectrogram images. Additionally, we aim to tackle the identification problem, determining which specific subject is performing each action. This introductory overview sets the stage for our exploration into the application of DL models in understanding and distinguishing activities and subjects within the dataset.

This goal is formalised using two research questions.

1.1 Research Question 1

This research question addresses the *classification problem*:

> How can deep learning image classification models be utilized to classify various actions carried out by different humans or animals into 6 classes, given their

TScIT 41, July 5, 2024, Enschede, The Netherlands

> respective Micro-Doppler Signatures, with a high reasonable accuracy?

The following sub-research questions would need to be answered, to answer the main question:

(1) Given the lack of data, which data augmentation techniques would be feasible to produce more data and prevent overfitting?
(2) From current literature, how are models trained to handle the image classification of Micro-Doppler Signatures?
(3) How can transfer learning techniques be utilized for this purpose to adapt models already trained on larger datasets?

1.2 Research Question 2

This research question addresses the *identification problem*:

> How can deep learning models be utilized to identify specific animals given their respective Micro-Doppler Signatures?

The following sub-research questions would be answered, to answer the main one:

(1) How can the high intra-class variance be reduced?
(2) How can the subject-specific features be extracted?

2 LITERATURE REVIEW

Multiple reviewed papers [20, 9, 5, 19, 23, 7, 21, 8, 22] show the most popular image classifiers with the model architecture and parameters used. Image classifiers have evolved from basic linear classifiers to complex deep learning models. Classifiers like Support Vector Machines (SVM) and k-Nearest Neighbors (k-NN) have been used for simple tasks, but Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) like (bi-directional) Long short-term memory models (LSTMs) have revolutionized this field of computer vision (CV), offering superior performance.

Moving more towards the specialised domain, there exists literature regarding image classification of spectrograms, hyperspectral images and Micro-Doppler radar data which contained ideal model parameters and results for proposed models [2, 3, 27, 43, 14, 37, 36, 6, 28, 29, 10, 25, 41]. Development in this specific domain lacks research with regards to activity; most spectrogram image classifiers are for Unmanned aerial vehicle (UAV) drones or audio samples.

Hence, transfer learning (TL), which is a technique where knowledge from a task or a model is used to enhance performance for a new but related idea, is a breakthrough to classifying MDS images for activity, which can also be done with images [12, 44, 33, 34, 39]. The knowledge obtained from hyperspectral and spectrogram image classification would be key for this purpose. Additionally, training a model from scratch may be too computationally expensive, so using a model with pre-trained weights would be crucial.

For the identification problem, literature exists with respect to person identification, with the main idea in [11] being preventing terrorist attacks and identifying illegal intrusions. There has been more person identification literature as well, such as [18, 4, 38, 31] with similar ideas of using CNNs and other DL methods to identify the person. To solve this problem in the past, techniques like Siamese networks [13] or Contrastive Learning [40] have proven to perform significantly well. However, there has not been significant research about animal identification using MDS.

To avoid over-fitting ML and DL models despite not having enough data, augmentation techniques need to be applied before inputting it into the model, so that it can learn more potential configurations of the same classes, hence avoiding misclassifying something it has not seen before [30]. For images, standard flipping, rotating or colour scale changes are usually done to alter the image. However, spectrogram images contain vital information; they cannot be treated just like any other image for augmentation. Hence, only certain image augments with restricted possibilities and parameters can be done [26, 17].

3 IMPLEMENTATION

Tables of the full model architectures are in section 6, the appendices of the paper. All models were developed in Python using *tensorflow.keras*.

3.1 Dataset

Some pre-processing of the dataset needed to be performed before jumping into the deep learning models for classification and identification. For each of the six classes of activities, two subjects were present. The humans were given the name "FirstHuman" and "SecondHuman". The cats were named "Bert" and "Turbo."

For each of these subjects, raw binary signal files were provided, which, after performing a series of signal processing algorithms, could extract the Micro-Doppler signature. However, every binary file had a different time range in which the phase-shifts of the activity were detected.

Along with these raw signal files, Comma-Separated Values (CSV) files representing the spectrogram were present, where the time range was already accounted for. Therefore, the MDS images were generated by using a simple script involving the *matplotlib.pyplot* library and saved for each activity and subject. There are approximately 37 images per activity per subject collected.

For the augmentation of the data, or simply expanding the data size, many image manipulation techniques like rotation or zooming would not work as they would distort important information from the images. Keeping this in mind, flipping from left to right is the only image manipulation technique which would work, as it would still capture the important phase shifts but just at a different time on the x-axis. Flipping up to down would duplicate data meaninglessly, as the images are symmetric in the x-axis. Just performing the horizontal flipping doubles the dataset size, resulting in the data distribution found in section 6, the appendix.

3.2 Classification

Before jumping right into classification, a copy of the current dataset was made for the task wherein all the images of the same activity were grouped together regardless of the subject, resulting in 148 images in each class, except for Cat Walking Towards, with 146. For all the classification models, the data with flipped and original images were loaded and the *image dataset from directory* function from *tensorflow.keras.preprocessing* was used with a 20% validation split and a seed of 123 with shuffle on, to ensure consistency in

the dataset for evaluation, yet, ensuring the same training data. Additionally, all the images are shaped to (128,128) and kept in RGB or RGBA.

After a review of the existing literature regarding image classifiers of spectrogram images, some models were shortlisted to be trained on the dataset in section 3.1.

The following models were trained from scratch:

(1) A Deep CNN Model
(2) A Bi-directional LSTM (Bi-LSTM)
(3) A D-CNN/Bi-LSTM Hybrid Model

As mentioned in section 2, Transfer learning can utilize pre-trained model weights from larger datasets, and some additional layers can be added to the model to then learn the detail about the relevant dataset [14]. Collecting a high amount of data is trickier for animal activity because the (human) data collectors might observe but not comprehend some unpredictable behaviour, which may result in a lower amount of data and/or a longer time necessary to collect data. Therefore, TL is a great option for this dataset. Additionally, it significantly reduce training time and computational resources and are powerful as feature extractors. The following models were trained using Transfer Learning:

(1) Transfer learning using A ResNet50 Model
(2) Transfer learning using An InceptionV3 Model
(3) Transfer learning using A MobileNetV2 Model

During training, these models are initialised with weights from the *imagenet* dataset, and the top layer for these models are not included. These models were imported from *tensorflow.keras.applications*.

The models which were trained from scratch will be presented below.

3.2.1 Deep-CNN Model. For this model's architecture, [10] was used as inspiration to create an activity classification model, since it also involves Micro-Doppler Signature image classification. This model was trained from scratch, but slight changes were made while implementing the model to enhance the results catered to the provided dataset. For example, the dataset was kept as RGB images as they are instead of converting it to grey-scale to avoid loss of important information which colours can show in spectrograms. One less layer was also used to reduce the model complexity a little since not much data was available in comparison to other studies; a complex model could result in overfitting by learning more than necessary about the training data. The final model architecture can be seen in Table 2. The model was compiled using the Stochastic Gradient Descent (SGD) optimizer with a learning rate of 0.001, sparse_categorical_crossentropy loss for 75 epochs. The changes made to this model compared to [10] could potentially alter the optimal parameters for the model in that study, but it is important to adjust the model in accordance to the dataset being used in this study. Specifically, the subject in this study involve animals, and a lower volume of data is available, both of which need adjusted parameters.

3.2.2 Bi-LSTM Model. LSTM stands for Long Short-Term Memory, and these networks are used to retain information and context in the network for longer. Bidirectional refers to the capability of processing the input data in both forward and backward directions, allowing the network to have access to both past (previous inputs) and future (later inputs) context when making predictions. To make this classifier model, inspiration was taken from [41], where Xception performed the best for feature extraction. Weights in the base feature extraction model were used from imagenet and the top layer was not included. A GlobalAveragePooling was then added. The in-built Xception pre-processing function was used and the images were reshaped into (32, 128, 128, 3) where 32 is the batch size, 3 represents RGB images and 128 represent the height and width of the images. Features were extracted for the train and test data and the labels were one-hot encoded. The output shape is (samples, time steps, features). The model extracts 2048 features with 1 time step, and the number of samples are 709 for the train set and 177 for the validation set. The Bi-LSTM decoder is inspired by the one used in [41], except for the input shape and the output number of classes. There were more optimization layers like dropout added, however, to avoid the model overfitting, by selecting random neurons to not be active during some training cycles, ensuring the model does not learn too quickly. Batch Normalization was also added to ensure the training is not too slow and the layer inputs are re-centered and re-scaled, stabilizing the training process and improving the model's ability to generalise, hence avoiding overfitting. Table 3 shows the architecture of this model. The SGD optimizer was used with a learning rate of 0.005 at 110 epochs.

3.2.3 DCNN/Bi-LSTM Hybrid Model. This classifier model had the same D-CNN architecture as the first classifier model. To integrate the Bidirectional LSTM aspect into this model, a Bidirectional LSTM layer was added before the classification Dense layer, but not before a reshaping layer to be compatible with the Bi-LSTM. This model helps to examine what difference an extra layer makes to a model, and further, checks what impact can the performance have with a Bi-LSTM aspect to a D-CNN model for the specific dataset used. Table 4 shows the model architecture used. This model was compiled and trained using the SGD optimizer with a learning rate of 0.001 for 120 epochs.

The models which were trained using Transfer Learning will be presented below.

3.2.4 ResNet-50 Model. Deep Neural Networks (DNNs) often learn the most complex information when the model has more layers. However, adding too many layers can be a problem, since gradients may uncontrollably diminish or explode. To counter this, ResNet models are used. To put it simply, they have "skip connections" which skips some layers when necessary. Given the complexity of Micro-Doppler Signatures, a ResNet model can help keep the gradient stable while the complex features of the spectrogram are still learned. Using transfer learning as well, a large volume of data would not be necessary as the weights from imagenet's dataset are pre-loaded. The base model is not set to trainable for this model for computational complexity reasons. The classification part of the model is very short, only having one dense layer and then the final output layer, which is where the model learns some details about the MDS. Table 5 formally shows the classifier structure. SGD with a 0.01 learning rate was used for model compilation, and it was run for 40 epochs.

3.2.5 InceptionV3 Model. In the case of InceptionV3, the base model is set to trainable, as the performance when the model was not trainable was poor with severe overfitting. The classification part of the model is, once again, very short, only having one dense layer and then the final output layer. However, this model used more optimization layers like Dropout. The model also uses L2 Regularisation, which is where a penalty equal to the sum of the squared values of the weights multiplied by the factor is added to the loss function. The penalty is useful to prevent overfitting. Table 6 formally shows the classifier structure. SGD with a 0.001 learning rate was used for model compilation, and it was run for 37 epochs. Additionally however, to improve the performance and reduce the overfitting, a learning rate scheduler was used, reducing the learning rate by a factor of 0.01 every 10 epochs, and a callback to reduce the learning rate on validation loss plateau by 0.1 with a patience of 5 epochs was also used.

3.2.6 MobileNetV2 Model. MobileNetV2 is a lot more lightweight than the other two pre-trained models. Structure-wise, it has been trained in a similar way to the ResNet50 model, but due to the lightweightedness, optimizations were made. One such optimization was the kernel initializer, set to normal, which means that the model initializes weights with values drawn from a truncated normal distribution. Table 7 shows the structure of the model. The model was trained for 40 epochs with a callback to reduce the learning rate on plateau of validation loss by a 0.1 factor with a patience of 5 epochs. SGD was again preferred as the optimizer, with a learning rate of 0.001 and a momentum of 0.9. SGD has the momentum property to help the model learn better and jump out of any minimas it falls into while training. A batch size of 16 was used here, whereas the other models used the default value of 32. This was done due to the model being lightweight, so that the model can generalise better.

3.3 Identification

The identification of a specific subject carrying out the action is a trickier problem to tackle because, although Micro-Doppler Signatures can be unique to both the subject and activity, the differences, including phase shift variations, are often minor, which makes it difficult to distinguish subjects, even for sophisticated deep learning methods [13]. Hence, the classes "Bert" and "Turbo" have a high intra-class variance [13]. To tackle this, a Siamese-like network needed to be employed. It was then considered to only use the data from actions which are similar to each other, making it easier to extract information about the cat's activities more accurately and efficiently. Therefore, the actions "Cat Walking By," "Cat Walking Towards" and "Cat Walking with Human" were used for this purpose.

The concept of the implemented network was inspired from [13], however re-using the actual model architecture from this literature would not result in ideal performances for this specific purpose. This is because the literature used 108,000 spectrogram images just for training purposes, meanwhile the dataset for Bert and Turbo used to train and validate this model is a mere 221.

The first step is to create pairs of these images, since the model accepts two images and finds out the difference between the two. The images are first loaded with their respective labels, Bert or Turbo. From the loaded 221 images, 438 pairs were made. These pairs were labelled on the basis of similarity; two images with of the same cat are categorised as a positive pair and two images with different cats are a negative pair. The positive pairs are made by pairing up images sequentially. For example, the pair (Bert1, Bert2) would be made with the label 1. Then the pair (Bert2, Bert3) would be made, and so on. Meanwhile, the negative pairs are made by pairing the same indices with each other, and then making a pair of the reverse combination. For example, the pair (Bert1, Turbo1) would be labelled with 0, and the pair (Turbo1, Bert1) will also be made with label 0. Making separate pairs for the reverse combinations ensures symmetric learning, so the model does not favor one order over the other, leading to a balanced learning process where both orders contribute equally to improving the model's distinction performance. This pairing strategy results in the number of pairs being almost double of the initial dataset, while also ensuring that the number of positive and negative pairs are similar, avoiding class imbalance. At the same time, every image (except for the first and last images of both Bert and Turbo) is part of two pairs per label. This method also makes image pairs with the same activity only, ensuring that the model is trained mainly on the differences between the subjects (same or different) and not on the differences arising from different activities. The pairs were split with a 20% validation ratio and a seed of 42.

It is important to note that the Siamese data does not use the flipped images, just the original ones which is done to avoid the possibility of pairing an original and flipped image together, which can induce bias in the model.

Moving on to the model architecture, the base network comprises of a 4-layer system, which is explained in more detail in Table 8.

Two instances of this base model are created so that they can share weights and learn together. One base network takes the first image of a specific pair as input and the other takes the second image of the pair. A lambda layer then calculates the Euclidean distance between these two images, which indicates the similarity of the two images. A basic contrastive loss function is used to minimize the distance between similar pairs and maximize the distance between dissimilar pairs. The formula for this loss function is given as:

$$L(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} \left[y_i \cdot \hat{y}_i^2 + (1 - y_i) \cdot \max(1 - \hat{y}_i, 0)^2 \right] \quad (1)$$

where:

- y_i: The true label for the i-th pair. It is 1 if the pair is similar (positive pair) and 0 if the pair is dissimilar (negative pair).
- $\hat{y}_i$: The predicted distance (or dissimilarity) between the two elements of the i-th pair.
- N: The total number of pairs.

The term $y_i \cdot \hat{y}_i^2$ in the equation aims to minimize $\hat{y}_i$, the predicted Euclidean distance, for similar pairs ($y_i = 1$). Ideally, $\hat{y}_i$ should be as small as possible, approaching zero, which signifies high similarity between the pair. On the other hand, the term $(1-y_i) \cdot \max(\text{margin} - \hat{y}_i, 0)^2$ penalizes dissimilar pairs ($y_i = 0$) where $\hat{y}_i$, the predicted distance, is less than 1. The penalty increases as $\hat{y}_i$ approaches zero, ensuring dissimilar pairs are adequately separated by at least the specified margin.

The overall loss is the mean of these contributions across all pairs, encouraging the model to reduce the distance for similar pairs and increase it for dissimilar pairs beyond the specified margin. The main model is then instantiated with the base models and the lambda layer, with the inputs being the two images of the pair and the output being the distance. The Adam optimizer with the default learning rate of 0.001 is used. The model is trained for 20 epochs with a batch size of 16.

The identification metrics require some pre-processing opposed to the classification models, since the output here is the Euclidean distance and not a class prediction yet. Therefore, a prediction function is created which take both the images as input and return the distance, which is then thresholded to 0.5 since it is a binary classification, whether the image pair is similar or dissimilar, to be precise.

4 RESULTS AND DISCUSSION

4.1 Classification

The models mentioned in section 3.2 were trained on on the dataset in section 3.1 according to the specifications for each model, and the results are being discussed in this section. The full results can be seen in section 6, which contains a table comparing the results of each model (Table 9) and confusion matrices for each model (Figures 1, 2, 3, 4, 5, 6)

The first essential metric which evaluates the models and informs us whether this model is suitable for the classification or not is validation accuracy. This is the accuracy of the model based on the validation data, so data that the model has not been trained on. As section 3.1 states, this is 20% of the total data.

However, just accuracy is not a strong enough metric, since any dataset could be imbalanced. Even though the training data used is balanced well, it is good practice to use additional metrics in general, since accuracy alone is not strong enough to depict what classes the model is good at predicting. Therefore, the f1 score is also used, which is the harmonic mean of precision and recall. The formulae for all of these are given below:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2}$$

where Precision measures the accuracy of a model in predicting a specific class, while recall measures the proportion of actual instances of a specific class that were correctly identified by the model. An overall f1-score is also computed, which is a weighted average of all the classes, where the weights denote the number of actual instances for each class.

Table 9 in the appendices, section 6 shows the results of the classification. It can be observed that the pre-trained models using transfer learning performed better than the models trained from scratch. Using the pre-trained weights was helpful for the computational power as well, as these models did not need to run for too many epochs before achieving satisfactory results. During the training process, it was also observed that the metrics for ResNet50 and MobileNetV2 did not fluctuate much for the first few epochs. This can be explained by the frozen layers; the base models were not trainable, therefore, it took a few epochs for the new layers on top to start learning.

Without techniques to combat overfitting like dropout, the Deep CNN model and the hybrid model struggled; 4 dropout layers were necessary to control the training process, ensuring the model was not overfitting, which means that the model was not learning details only specific to the training data, but details which it can generalise for data it has not seen or trained with. In general, models trained from scratch have to learn all features, and with the amount of data available for this study, the models would be bound to overfit, which can be seen by a huge difference between the train and validation accuracies. The Bi-LSTM model had a pre-trained model (Xception) for feature extraction, and it was a smaller model, so having too many dropout layers was unnecessary. BatchNormalization was also used to normalize layer outputs and ensure the training is stable. The Bi-LSTM needed 2 layers to ensure the training process was quick and still prevents overfitting by providing a regularisation effect. ResNet has a natural combat to overfitting by using skip connection, so it did not have any regularization technique, but InceptionV3 and MobileNetV2, being deeper models, needed to employ this for some layers. It adds a penalty to the loss depending on the type of regularization used. L1 regularization adds the sum of the absolute values of the weights to the loss function, while L2 regularization adds the sum of the squared values of the weights to the loss function.

Though the gap between training and validation accuracies for certain models may raise suspicions of overfitting, it is important to consider the limited data volume used for training. Keeping this in mind, the observed difference is not significant enough to definitively label the model as overfitting.

InceptionV3 performed the best. The reason for this could be that pre-trained weights were being used, but the model was also initially unfrozen. This combination often results in an overfitting model, but with a careful consideration of the hyper-parameters (optimizer, learning rate and callbacks in this case) used, the model can generalise properly..

The hybrid model did not perform too well with the given dataset, compared to the individual DCNN and BiLSTM models. There are a few reasons for this. Adding this extra BiLSTM layer increases the model's complexity, which makes it harder for the model to converge during training. As the results show, it managed to learn the least from the training set compared to the other models. With a more complex model with different kinds of layers, more hyperparameter tuning needs to be done to achieve a better performance. Additionally, it could be that the Reshape before passing the information to the BiLSTM layer could be a reason for the loss of some vital information, which is why the results are not as high as the hybrid model's individual counterparts. Finally, a DCNN is also suited better for extracting the features of images, which the BiLSTM layer might have hindered.

To explain the differences in the F1-score for each class, the "Eating" activity was the only class the most images where the position of the phase shifts were not quite centered, which might explain the high scores. "WalkingTowards" also had the most distinguishable pattern as observed by the human eye, which might also explain how it achieved a 100% for one class. Some images in "Jumping" and "WalkingBy" also had shared patterns which can explain the similarities in metrics for some models, and also the confusion matrices for

some models. Some misclassifications, as observed by the confusion matrices, were also caused because of the "WalkingWithHuman" class, which has aspects of both cats walking and humans walking.

4.2 Identification

The evaluation of the Siamese network is inherently different compared to the metrics of the classification network. For starters, the classification networks take an image as input and predict the class of the activity. This model takes a pair of images as input and predicts whether the images are similar or not. This would, for example, be done by passing in two images of Bert and seeing if the model correctly returns similar or incorrectly returns dissimilar, as the output. The same concept of F1-score from section 4.1 is used for the identification metrics too. The F1-score for dissimilar pairs is 84%, indicating a strong ability to correctly identify pairs of images that are not similar. Meanwhile, the F1-score for similar pairs is 81% which demonstrates that the model performs well in identifying similar pairs as well. The results are formally shown by Table 10 and Figure 7 in section 6.

Perhaps, the scores would be higher if more animal MDS image data was available, using the same concept from [13], but due to the complications with collecting animal-based data, this remains a question for the future. Additionally, due to a lack of hardware resources, a model with a greater amount of pairs, where each image is paired with every other image, creating $M*(N)^2$ pairs with labels, could not be effectively tested, which is also something which should be looked at in the future, where N represents the total number of images in the dataset per activity, and M represent the number of activities.

5 CONCLUSION

With this study, we compared different image classifier models and found that pre-trained models with Transfer Learning, especially InceptionV3, outperformed models being trained from scratch. A reason for this performance difference is the lack of volume of MDS data for for animal activity, which is due to the fact that collecting animal data is more tricky since their movements are unpredictable and they may not know what they should do. For example, collecting jumping data from a human and a cat is different, since humans generally understand that they must jump, but cats need to be trained to know what action should be performed.

Research Question 1 from section 1.1 was answered by firstly creating more data by flipping left to right, which is the time scale, so it does not affect the information about the activity. Supervised models from previous literature were then reviewed and examined on the dataset in this study with adjustments to layers and optimizations due to a lack of data. Transfer Learning approaches using common models and the imagenet dataset's weights were also examined to find that they slightly outperform the models trained from scratch, because they have initial weights from a bigger dataset and do not need data from the smaller dataset from this study for these weights.

Research Question 2 from section 1.2 was answered by using a supervised Siamese network with contrastive loss to determine whether two images were alike or not. Initially, pairs of images were created from the different images of Bert and Turbo, the two cats to be identified. These were given labels for similar or dissimilar, and a backbone network was created. The Siamese model contains the backbone network with two inputs, each image from respective pairs, and outputs a Euclidean distance showing similarity. The model was then evaluated by setting the threshold of the output distance to 0.5 since it is a binary classification of the two cats.

This study hopes to have provided a breakthrough into classification of actions and identification of subjects using animal Micro-Doppler Signature images. With a greater amount of animal data, this can be looked into with more detail using the model architectures from this study with adjustments to the model complexity depending on the dataset size.

6 APPENDICES

During the preparation of this work, the author used the tool ChatGPT 4o and 3.5 to assist with some model architecture, and LaTeXfor this document. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the work.

Activity	**Number of Images**	
	Subject 1	**Subject 2**
Cat Eating	74	74
Cat Jumping	74	74
Cat Walking	74	74
Cat Walking Towards	74	72
Cat Walking with Human	74	74
Human Walking	74	74

Table 1. Activity and Number of Images for Each Subject after left to right flips

Layer Type	**Parameters**
Resize Image	(128,128,3)
Conv2D	32 filters, (3,3) kernel, ReLU
MaxPooling2D	(2,2)
Dropout	0.2
Conv2D	64 filters, (3,3) kernel, ReLU
MaxPooling2D	(2,2)
Dropout	0.3
Conv2D	128 filters, (3,3) kernel, ReLU
MaxPooling2D	(2,2)
Dropout	0.4
Flatten	
Dense	128, ReLU
Dropout	0.5
Dense	6, Softmax
Total Trainable Parameters: 3,305,414	

Table 2. Deep-CNN Model Architecture

Layer Type	Parameters
Resize Image	(128,128,3)
Bidirectional LSTM	512 filters, return_sequences=True
Dropout	0.5
BatchNormalization	
Bidirectional LSTM	256 filters
Dropout	0.5
BatchNormalization	
Dense	256, ReLU
Dense	6, Softmax
Trainable Parameters: 13,249,286	

Table 3. Bi-LSTM Model Architecture

Layer Type	Parameters
Resize Image	(128,128,3)
Conv2D	32 filters, (3,3) kernel, ReLU
MaxPooling2D	(2,2)
Dropout	0.2
Conv2D	64 filters, (3,3) kernel, ReLU
MaxPooling2D	(2,2)
Dropout	0.3
Conv2D	128 filters, (3,3) kernel, ReLU
MaxPooling2D	(2,2)
Dropout	0.4
Flatten	
Dense	128, ReLU
Reshape	(-1, 128)
Bidirectional LSTM	64 filters, return_sequences=False
Dropout	0.5
Dense	6, Softmax
Total Trainable Parameters: 3,404,678	

Table 4. D-CNN/Bi-LSTM Hybrid Model Architecture

Layer/Model Type	Parameters
Resize Image	(128,128,3)
Resnet50	imagenet weights, no top layer
The Base Model	trainable = False
GlobalAveragePooling2D	
Dense	512, ReLU
Dense	6, Softmax
Total parameters: 24,639,878	
Trainable parameters: 1,052,166	

Table 5. ResNet50 Model Architecture

Layer/Model Type	Parameters
Resize Image	(128,128,3)
InceptionV3	imagenet weights, no top layer
The Base Model	trainable = True
GlobalAveragePooling2D	
Dropout	0.4
Dense	1024, ReLU
Kernel Regularizer	L2(0.01)
BatchNormalization	
Dropout	0.5
Dense	6, Softmax
Total parameters: 23,911,206	
Trainable parameters: 23,874,726	

Table 6. InceptionV3 Model Architecture

Layer/Model Type	Parameters
Resize Image	(128,128,3)
MobileNetV2	imagenet weights, no top layer
The Base Model	trainable = False
GlobalAveragePooling2D	
Dropout	0.3
Dense	512, ReLU
Kernel Regularizer	L1(0.01)
Kernel Initializer	Normal
BatchNormalization	
Dropout	0.3
Dense	6, Softmax
Total parameters: 2,918,982	
Trainable parameters: 659,974	

Table 7. MobileNetV2 Model Architecture

Layer/Model Type	Parameters
Resize Image	(128,128,3)
Conv2D	62 filters, (3,3), ReLU
MaxPooling2D	(2,2)
Conv2D	128 filters, (3,3), ReLU
MaxPooling2D	(2,2)
Conv2D	256 filters, (3,3), ReLU
MaxPooling2D	(2,2)
Conv2D	512 filters, (3,3), ReLU
Flatten	
Dense	512 filters, ReLU
Dense	256 filters, ReLU
Trainable parameters: 18,460,032	

Table 8. Siamese Base Model Architecture

Model	F1-Score							Accuracy	
	Eating	HumanWalking	Jumping	WalkingBy	WalkingTowards	WalkingWithHuman	Overall	Train	Validation
DCNN	96%	81%	82%	82%	88%	87%	**87%**	90%	**87%**
BiLSTM	97%	87%	82%	83%	88%	80%	**86%**	91%	**86%**
Hybrid	89%	89%	74%	75%	90%	93%	**84%**	86%	**84%**
Resnet	96%	91%	95%	88%	93%	86%	**92%**	96%	**91%**
Inception	97%	88%	92%	86%	95%	96%	**93%**	96%	**93%**
MobileNet	94%	81%	91%	89%	100%	80%	**89%**	92%	**89%**

Table 9. Classification Performance of Various Models

Class	F1-score
Dissimilar pairs	84%
Similar pairs	81%
Weighted Average	**83%**

Table 10. identification Performance of the Proposed Siamese Network

Fig. 1. Deep-CNN Model Confusion Matrix

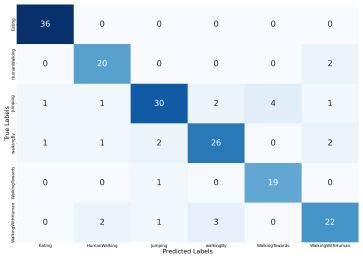

Fig. 2. BiLSTM Model Confusion Matrix

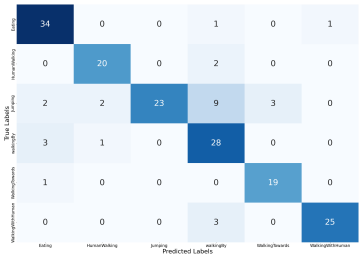

Fig. 3. Deep-CNN BiLSTM Hybrid Model Confusion Matrix

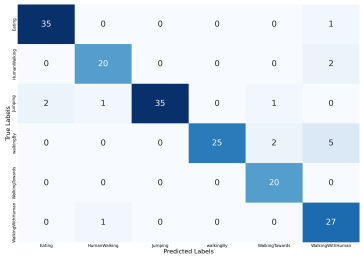

Fig. 4. ResNet50 Model Confusion Matrix

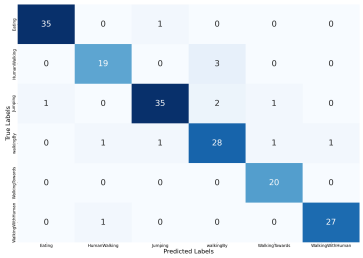

Fig. 5. Inception3 Model Confusion Matrix

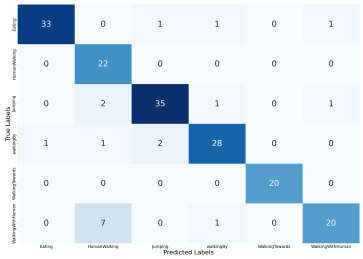

Fig. 6. MobileNetV2 Model Confusion Matrix

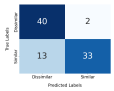

Fig. 7. Siamese identification Confusion Matrix

REFERENCES

[1] A. Ahmad, J. C. Roh, D. Wang, and A. Dubey. Vital signs monitoring of multiple people using a FMCW millimeter-wave sensor. In *2018 IEEE Radar Conference (RadarConf18)*. 2018 IEEE Radar Conference (RadarConf18), pages 1450–1455, Apr. 2018. DOI: 10.1109/RADAR.2018.8378778. URL: https://ieeexplore.ieee.org/abstract/document/8378778. ISSN: 2375-5318.

[2] A. Azab and M. Khasawneh. MSIC: malware spectrogram image classification. *IEEE Access*, 8:102007–102021, 2020. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2999320. URL: https://ieeexplore.ieee.org/document/9104999/.

[3] V. Boddapati, A. Petef, J. Rasmusson, and L. Lundberg. Classifying environmental sounds using image recognition networks. *Procedia Computer Science*. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France, 112:2048–2056, Jan. 1, 2017. ISSN: 1877-0509. DOI: 10.1016/j.procs.2017.08.250. URL: https://www.sciencedirect.com/science/article/pii/S1877050917316599.

[4] P. Cao, W. Xia, M. Ye, J. Zhang, and J. Zhou. Radar-ID: human identification based on radar micro-doppler signatures using deep convolutional neural networks. *IET Radar, Sonar & Navigation*, 12(7):729–734, 2018. ISSN: 1751-8792. DOI: 10.1049/iet-rsn.2017.0511. URL: https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-rsn.2017.0511.

[5] S. Y. Chaganti, I. Nanda, K. R. Pandi, T. G. Prudhvith, and N. Kumar. Image classification using SVM and CNN. In *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*. 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), pages 1–5, Mar. 2020. DOI: 10.1109/ICCSEA49143.2020.9132851. URL: https://ieeexplore.ieee.org/abstract/document/9132851.

[6] M. Chakraborty, H. C. Kumawat, S. V. Dhavale, and A. B. Raj A. Application of DNN for radar micro-doppler signature-based human suspicious activity recognition. *Pattern Recognition Letters*, 162:1–6, Oct. 1, 2022. ISSN: 0167-8655. DOI: 10.1016/j.patrec.2022.08.005. URL: https://www.sciencedirect.com/science/article/pii/S0167865522002434.

[7] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao. Review of image classification algorithms based on convolutional neural networks. *Remote Sensing*, 13(22):4712, Jan. 2021. ISSN: 2072-4292. DOI: 10.3390/rs13224712. URL: https://www.mdpi.com/2072-4292/13/22/4712. Number: 22 Publisher: Multidisciplinary Digital Publishing Institute.

[8] P. Dhruv and S. Naskar. Image classification using convolutional neural network (CNN) and recurrent neural network (RNN): a review. In D. Swain, P. K. Pattnaik, and P. K. Gupta, editors, *Machine Learning and Information Processing*, pages 367–381, Singapore. Springer, 2020. ISBN: 9789811518843. DOI: 10.1007/978-981-15-1884-3_34.

[9] T. Guo, J. Dong, H. Li, and Y. Gao. Simple convolutional neural network on image classification. In *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*. 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), pages 721–724, Mar. 2017. DOI: 10.1109/ICBDA.2017.8078730. URL: https://ieeexplore.ieee.org/abstract/document/8078730.

[10] S. Hassan, X. Wang, S. Ishtiaq, N. Ullah, A. Mohammad, and A. Noorwali. Human activity classification based on dual micro-motion signatures using interferometric radar. *Remote Sensing*, 15(7):1752, Jan. 2023. ISSN: 2072-4292. DOI: 10.3390/rs15071752. URL: https://www.mdpi.com/2072-4292/15/7/1752. Number: 7 Publisher: Multidisciplinary Digital Publishing Institute.

[11] Y. He, H. Guo, X. Zhang, R. Li, Y. Lang, and Y. Yang. Person identification based on fine-grained micro-doppler signatures and UWB radar. *IEEE Sensors Journal*, 23(18):21421–21432, Sept. 2023. ISSN: 1558-1748. DOI: 10.1109/JSEN.2023.3299558. URL: https://ieeexplore.ieee.org/abstract/document/10205518. Conference Name: IEEE Sensors Journal.

[12] M. Hussain, J. J. Bird, and D. R. Faria. A study on CNN transfer learning for image classification. In A. Lotfi, H. Bouchachia, A. Gegov, C. Langensiepen, and M. McGinnity, editors, *Advances in Computational Intelligence Systems*, pages 191–202, Cham. Springer International Publishing, 2019. ISBN: 978-3-319-97982-3. DOI: 10.1007/978-3-319-97982-3_16.

[13] S. Jia, S. Jiang, Z. Lin, M. Xu, W. Sun, Q. Huang, J. Zhu, and X. Jia. A semisupervised siamese network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–17, 2022. ISSN: 1558-0644. DOI: 10.1109/TGRS.2021.3116138. URL: https://ieeexplore.ieee.org/abstract/document/9565215. Conference Name: IEEE Transactions on Geoscience and Remote Sensing.

[14] K. Jung, J.-I. Lee, N. Kim, S. Oh, and D.-W. Seo. Classification of space objects by using deep learning with micro-doppler signature images. *Sensors*, 21(13):4365, Jan. 2021. ISSN: 1424-8220. DOI: 10.3390/s21134365. URL: https://www.mdpi.com/1424-8220/21/13/4365. Number: 13 Publisher: Multidisciplinary Digital Publishing Institute.

[15] Y. Kamat and S. Nasnodkar. Advances in technologies and methods for behavior, emotion, and health monitoring in pets. *Applied Research in Artificial Intelligence and Cloud Computing*, 1(1):38–57, Dec. 10, 2018. URL: https://researchberg.com/index.php/araic/article/view/158.

[16] T. Kiuru, M. Metso, S. Jardak, P. Pursula, J. Häkli, M. Hirvonen, and R. Sepponen. Movement and respiration detection using statistical properties of the FMCW radar signal. In *2016 Global Symposium on Millimeter Waves (GSMM) & ESA Workshop on Millimetre-Wave Technology and Applications*. 2016 Global Symposium on Millimeter Waves (GSMM) & ESA Workshop on Millimetre-Wave Technology and Applications, pages 1–4, June 2016. DOI: 10.1109/GSMM.2016.7500331. URL: https://ieeexplore.ieee.org/abstract/document/7500331.

[17] C.-E. Kuo, G.-T. Chen, and P.-Y. Liao. An EEG spectrogram-based automatic sleep stage scoring method via data augmentation, ensemble convolution neural network, and expert knowledge. *Biomedical Signal Processing and Control*, 70:102981, Sept. 2021. ISSN: 17468094. DOI: 10.1016/j.bspc.2021.102981. URL: https://linkinghub.elsevier.com/retrieve/pii/S1746809421005784 (visited on 05/18/2024).

[18] Y. Lang, Q. Wang, Y. Yang, C. Hou, Y. He, and J. Xu. Person identification with limited training data using radar micro-doppler signatures. *Microwave and Optical Technology Letters*, 62(3):1060–1068, 2020. ISSN: 1098-2760. DOI: 10.1002/mop.32125. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/mop.32125.

[19] H. Lee and H. Kwon. Going deeper with contextual CNN for hyperspectral image classification. *IEEE Transactions on Image Processing*, 26(10):4843–4855, Oct. 2017. ISSN: 1941-0042. DOI: 10.1109/TIP.2017.2725580. URL: https://ieeexplore.ieee.org/abstract/document/7973178. Conference Name: IEEE Transactions on Image Processing.

[20] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen. Medical image classification with convolutional neural network. In *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*. 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), pages 844–848, Dec. 2014. DOI: 10.1109/ICARCV.2014.7064414. URL: https://ieeexplore.ieee.org/abstract/document/7064414.

[21] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson. Deep learning for hyperspectral image classification: an overview. *IEEE Transactions on Geoscience and Remote Sensing*, 57(9):6690–6709, Sept. 2019. ISSN: 1558-0644. DOI: 10.1109/TGRS.2019.2907932. URL: https://ieeexplore.ieee.org/document/8697135. Conference Name: IEEE Transactions on Geoscience and Remote Sensing.

[22] Q. Liu, F. Zhou, R. Hang, and X. Yuan. Bidirectional-convolutional LSTM based spectral-spatial feature learning for hyperspectral image classification. *Remote Sensing*, 9(12):1330, Dec. 2017. ISSN: 2072-4292. DOI: 10.3390/rs9121330. URL: https://www.mdpi.com/2072-4292/9/12/1330. Number: 12 Publisher: Multidisciplinary Digital Publishing Institute.

[23] D. Lu and Q. Weng. A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing*, 28(5):823–870, Mar. 1, 2007. ISSN: 0143-1161. DOI: 10.1080/01431160600746456. URL: https://doi.org/10.1080/01431160600746456.

[24] W. Lv, W. He, X. Lin, and J. Miao. Non-contact monitoring of human vital signs using FMCW millimeter wave radar in the 120 GHz band. *Sensors (Basel, Switzerland)*, 21(8):2732, Apr. 13, 2021. ISSN: 1424-8220. DOI: 10.3390/s21082732. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8070581/.

[25] P. Mandal, L. P. Roy, and S. K. Das. Flying objects classification based on micro–doppler signature data from UAV borne radar. *IEEE Geoscience and Remote Sensing Letters*, 21:1–5, 2024. ISSN: 1558-0571. DOI: 10.1109/LGRS.2024.3354973. URL: https://ieeexplore.ieee.org/document/10401166/similar#similar (visited on 05/05/2024). Conference Name: IEEE Geoscience and Remote Sensing Letters.

[26] Z. Mushtaq, S.-F. Su, and Q.-V. Tran. Spectral images based environmental sound classification using CNN with meaningful data augmentation. *Applied Acoustics*, 172:107581, Jan. 15, 2021. ISSN: 0003-682X. DOI: 10.1016/j.apacoust.2020.107581. URL: https://www.sciencedirect.com/science/article/pii/S0003682X2030685X (visited on 05/18/2024).

[27] L. Nanni, A. Rigo, A. Lumini, and S. Brahnam. Spectrogram classification using dissimilarity space. *Applied Sciences*, 10(12):4176, Jan. 2020. ISSN: 2076-3417. DOI: 10.3390/app10124176. URL: https://www.mdpi.com/2076-3417/10/12/4176. Number: 12 Publisher: Multidisciplinary Digital Publishing Institute.

[28] İ. Özer, S. B. Efe, and H. Özbay. CNN/bi-LSTM-based deep learning algorithm for classification of power quality disturbances by using spectrogram images. *International Transactions on Electrical Energy Systems*, 31(12):e13204, 2021. ISSN: 2050-7038. DOI: 10.1002/2050-7038.13204. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/2050-7038.13204.

[29] J. Park and J.-S. Park. Classification of small drones using low-uncertainty micro-doppler signature images and ultra-lightweight convolutional neural network. *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*, 33:2979–2994, 2024. ISSN: 1941-0042. DOI: 10.1109/TIP.2024.3388895.

[30] L. Perez and J. Wang. The effectiveness of data augmentation in image classification using deep learning, Dec. 13, 2017. arXiv: 1712.04621[cs]. URL: http://arxiv.org/abs/1712.04621.

[31] X. Qiao, Y. Feng, T. Shan, and R. Tao. Person identification with low training sample based on micro-doppler signatures separation. *IEEE Sensors Journal*, 22(9):8846–8857, May 2022. ISSN: 1558-1748. DOI: 10.1109/JSEN.2022.3162590. URL: https://ieeexplore.ieee.org/abstract/document/9743392. Conference Name: IEEE Sensors Journal.

[32] S. Ramachandran, L. H. Palivela, V. Vijayakumar, V. Subramaniyaswamy, J. Abawajy, and L. Yang. An intelligent system to detect human suspicious activity using deep neural networks. *Journal of Intelligent & Fuzzy Systems*, 36(5):4507–4518, 2019. ISSN: 10641246. URL: http://ezproxy2.utwente.nl/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=136448643&site=ehost-live.

[33] Q. Shi, Y. Zhang, X. Liu, and K. Zhao. Regularised transfer learning for hyperspectral image classification. *IET Computer Vision*, 13(2):188–193, 2019. ISSN: 1751-9640. DOI: 10.1049/iet-cvi.2018.5145. URL: https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-cvi.2018.5145.

[34] H. Sun, S. Liu, S. Zhou, and H. Zou. Unsupervised cross-view semantic transfer for remote sensing image classification. *IEEE Geoscience and Remote Sensing Letters*, 13(1):13–17, Jan. 2016. ISSN: 1558-0571. DOI: 10.1109/LGRS.2015.2491605. URL: https://ieeexplore.ieee.org/abstract/document/7314893. Conference Name: IEEE Geoscience and Remote Sensing Letters.

[35] D. Tahmoush. Review of micro-doppler signatures. *IET Radar, Sonar & Navigation*, 9(9):1140–1146, 2015. ISSN: 1751-8792. DOI: 10.1049/iet-rsn.2015.0118. URL: https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-rsn.2015.0118.

[36] T.-H. Tan, J.-H. Tian, A. K. Sharma, S.-H. Liu, and Y.-F. Huang. Human activity recognition based on deep learning and micro-doppler radar data. *Sensors*, 24(8):2530, Jan. 2024. ISSN: 1424-8220. DOI: 10.3390/s24082530. URL: https://www.mdpi.com/1424-8220/24/8/2530. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute.

[37] W. Taylor, K. Dashtipour, S. A. Shah, A. Hussain, Q. H. Abbasi, and M. A. Imran. Radar sensing for activity classification in elderly people exploiting micro-doppler signatures using machine learning. *Sensors*, 21(11):3881, Jan. 2021. ISSN: 1424-8220. DOI: 10.3390/s21113881. URL: https://www.mdpi.com/1424-8220/21/11/3881. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute.

[38] B. Vandersmissen, N. Knudde, A. Jalalvand, I. Couckuyt, A. Bourdoux, W. De Neve, and T. Dhaene. Indoor person identification using a low-power FMCW radar. *IEEE Transactions on Geoscience and Remote Sensing*, 56(7):3941–3952, July 2018. ISSN: 0196-2892, 1558-0644. DOI: 10.1109/TGRS.2018.2816812. URL: https://ieeexplore.ieee.org/document/8333730/.

[39] H. Wang, F. Nie, H. Huang, and C. Ding. Dyadic transfer learning for cross-domain image classification. In *2011 International Conference on Computer Vision*. 2011 International Conference on Computer Vision, pages 551–556, Nov. 2011. DOI: 10.1109/ICCV.2011.6126287. URL: https://ieeexplore.ieee.org/abstract/document/6126287. ISSN: 2380-7504.

[40] P. Wang, K. Han, X.-S. Wei, L. Zhang, and L. Wang. Contrastive learning based hybrid networks for long-tailed image classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 943–952, 2021. URL: https://openaccess.thecvf.com/content/CVPR2021/html/Wang_Contrastive_Learning_Based_Hybrid_Networks_for_Long-Tailed_Image_Classification_CVPR_2021_paper.html.

[41] Y. Wang, M. Zhang, R. Wu, H. Wang, Z. Luo, and G. Li. Speech neuromuscular decoding based on spectrogram images using conformal predictors with bi-LSTM. *Neurocomputing*, 451:25–34, Sept. 2021. ISSN: 09252312. DOI: 10.1016/j.neucom.2021.03.025. URL: https://linkinghub.elsevier.com/retrieve/pii/S092523122100391X (visited on 05/20/2024).

[42] M. Zhou, Y. Liu, S. Wu, C. Wang, Z. Chen, and H. Li. A novel scheme of high-precision heart rate detection with a mm-wave FMCW radar. *IEEE Access*, 11:85118–85136, 2023. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3303335. URL: https://ieeexplore.ieee.org/abstract/document/10210540. Conference Name: IEEE Access.

[43] J. Zhu, H. Chen, and W. Ye. A hybrid CNN–LSTM network for the classification of human activities based on micro-doppler radar. *IEEE Access*, 8:24713–24720, 2020. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2971064. URL: https://ieeexplore.ieee.org/document/8978926/.

[44] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, Jan. 2021. ISSN: 1558-2256. DOI: 10.1109/JPROC.2020.3004555. URL: https://ieeexplore.ieee.org/document/9134370.